



# PECTKUM

■ világ ■ 14. rész ■



49,- Ft





# RECENTRA

Az itt látható SPECTRUM 48K, SPECTRUM 128K (S107, S108) kollekciók is megrendelhetők a SPECTRUM VILÁG címén keresztül.

Egy kazetta ára ÁFA-val és postaköltséggel együtt: 300,- Ft.

Kérjük megrendeléseikben a kazetták sorszámát  *pontosan megnevezni!*

S101/A: Quackshot - Southern Belle - Beach Head II. - Commando - Space Shuttle - Dangerous Garden - Formula #1

B: Time Gate - Ant Attack - Mined Out - Humpty in the Garden - Halls of the Things - Dragonfire

S102/A: Ghostbusters - Rally Driver - The Neverending Story (+ part 3) - Impossible

B: Metabolite - Highway - Zombie 2 - Fantastic Voyages - Hellfire

S103/A: Bounces - Boulder Dash III - Think - Paws - Soul of a Rebel - Planet 1

B: Planets 2 - Aliquid Simplex - Wrangler - Lorio Tom - New Oylon Attack - Astro

S104/A: Sorderon's Shadow - Robin of Sherwood - Grumpy Gumphrey Supersleuth - Frenzy - Cosmazoigs - The Snowman - Xadom - Space Intruders

B: Green Beret - Twister - V the Game - Benny - World Cup Football - IS-Cyrus

S105/A: Battle Ships - Gnome Ranger I-II-III - Dream Warrior - Pogostick Olympics

B: Rex Hard - Muggins the Spaceman - Every Second Counts (+ part 5) - Blood

S106/A: Star Wars Basketball (part 1-4) - Ole Toro - 20 Stanzas - Hard Guy - Molar

B: Maul - Fantasy - Cybernoid 2 (The Revenge) - Battle Valley - Rogue - Kemsbu - Skateboard Kidz - Thunderblade (D)

S107 (128K)/A: Platoon 128 - Gyrzor 128 - Zub 128

B: Shanghai Karate I-II-III - Buggy Boy 128 - Joe Blade 128

S108 (128K)/A: Stiffup & Co. I-II, 128 - Death Wish-3 128 - Renegade 128

B: Combat School 128 - The Tube 128 - Ace-2 128 - Robin of the Wood 128

S109/A: MICRO PROLOG PACKAGE 'A': Main (3), Simple (25), Simtrace (13),

Erptrap (6), and (3), Simshow (2), Program (8), Deltap (3), Erptrap (3), Erptrap

(6), T. and (3), Tasword Two H - Tasword Two 2H - Tasword Two M -

Taswide - Taswide H - Tasword Two H - Tasword Two 2H - Tasword Two M -

B: MICRO PROLOG PACKAGE 'B': Main (3), Micro (17), Erptrap (6), Told (3),

Mischow (2), Editor (20), Modules (4), Traps (9), Spritrac (3), Erptrap (6),

Transprint (+ microdrive extension) - LERM Toolkits: Bytes Movie, Advanced

Header Reader, Header Creator, Program Analysis: Games Designer

S110/A: Countries of the World - Education Package: Reactivity, Text, Trigonometry,

Transform, Statistics, Odds, Text, Formulae, Circuits, Light;

B: Education Package (folyt.): Machine, Text; - Dynamic Programming - Statistics -

Address Manager

Megjegyzés: \* - felhasználói programok



# COMMODORE 64 RECENTRA

Az itt látható COMMODORE 64 kollekciók is megrendelhetők a SPECTRUM VILÁG címén keresztül.

Egy kollekció ára ÁFA-val és postaköltséggel együtt:

kazettán: 250,- Ft, lemezen: 400,- Ft.

Kérjük megrendeléseikben a kollekciók sorszámát  *pontosan megnevezni!* (lemezen történő rendelés esetén a kollekciót 'M' jelzéssel lássuk el - pl. C102M)

C101/A: THE LAST NINJA - The Gardens - The Wilderness - Palace Gardens

B: THE LAST NINJA - The Dungeons - The Palace - The Inner Sanctum

C102/A: THE LAST NINJA 2 - Central Park - The Street - The Sewers

C103/A: C: The Basement - The Office - The Mansion - Final Battle

B: Movie Monst. Space - Nexus

C104/A: Legend of the Amazon Warrior - Rat Race - Time Runner

B: Tarzan - Minit Man - Shades - Geisterjäger

C105/A: Labyrinth - Omega Mission - Spy Trek - Delight

B: Dig Dug - Forbidden Forest - Rocket Roger - Anticipital

C106/A: Trashman - Star Race - Hopping Elf - Starpost

C107/A: Football Frenzy - Flying Shark - Microgo 1 - Lemonade - Zauberschloss

B: Freddy Hardest I-II - Rygar - Action Force

C108/A: Pinatium - Spore - Wacky Worlds - Super Soccer - Echolon

C109/A: Evening Star - Scramble - Clean Up Service - The Dark Side

C110/A: Graphics BASIC - Assembler 48K2 - Shoot em Construction Kit

B: Front 32 (D) - Viper - Shoot em Construction Kit

B: English Education (part 1-16)

Megjegyzés: \* - felhasználói programok

A SpV 12. részében megjelent keresztjelvény szerencsés nyertesel:

B.G. - Budapest, E.L. - Debrecen, H.Z. - Budapest, P.E. - Budapest, V.J. - Szombathely

A SpV 13. részében megjelent keresztjelvény helyes megfejtései:

Vtazs.1.: ROMANTIC ROBOT, 27.; SICOR, 83.; JACK THE NIPPER,

Fig.4.: ACE, 8.; CAMBRIDGE, 14.; MELBOURNE DRAW, 22.;

CHEQUERED FLAG



Bizonyára már mindenki éhesen várja az információt: **'végre megjelent a SAM!'** pedig erre még egy kicsit várni kell, egészen pontosan a Miles Gordon Technology tervezői azt nyilatkozták, hogy 1989. első negyedévében már hírt adnak a premierről. Ami változott az eredeti információhoz képest az elsődlegesen a gép 100,- Font alatti induló ára, most 130-140,- Fontra tartják a készülök szuperklónt, s ez talán annak is köszönhető, hogy a gép Japánban fog készülni, bár a Japán készülékek mindig is olcsóságukról tanúskodtak, úgy érezzük, hogy a gépre szabott új ár nem is olyan borzasztó.

Miközben pezseg a piac, s mindenki tágra nyílt szemmel figyeli a változásokat, az AMSTRAD cég dobta be magát egy meghökkentő húzással: **'MEGJELENT AZ ÚJ SINCLAIR GÉP!!!'**

Azt úgy érezzük nem kell ecsetelnünk, hogy az új SINCLAIR gépnek SIR CLIVE SINCLAIR-hez semmi köze a névazonosságon kívül, hiszen az AMSTRAD a SINCLAIR név használatát is kizárólagosan megvette. Mégis mi vezette a céget egy ilyen dologra, hogy új IBM PC kompatibilis gépet SINCLAIR PC-200-nak keresztelje, ezt talán csak az ötlet szülőatyja tudná megmondani, bár úgy érezzük ennek elsősorban üzleti megfontolású oka van.

## Nos mit is tud az új SINCLAIR?

Természetesen nem kompatibilis egyik Spectrum verzióval sem, annál inkább IBM PC kompatibilis. Kisméretű, fekete műanyag dobozba építették, úgy mint hajdanán elődjait. Az alapgép és a billentyűzet egy egységet alkot, a gép hűtését belső ventilátor biztosítja. Klaviatúrája 102 billentyűt tartalmaz, a billentyűk elhelyezkedése könnyen áttekinthető, a szabványos írógépbillentyűzet köré csoportosulnak a kiegészítő funkciók billentyűi (Kurzor billentyűk, CTRL és ALT, nemzetközi tízes billentyűzet, Page Up, Page Down, Home, Delete, Insert, stb.), amelyek az IBM PC/AT standard billentyűzetének felelnek meg. Ja és a 10 funkció billentyű (F1-F10) szintén rendelkezésre áll. A NUM LOCK és SCROLL LOCK billentyűk funkcióját kis zöld LED-ek is támogatják.

A számítógép központi egysége egy 16 bites 8 MHz-en keteygő INTEL 8086 típusú mikroprocesszor. Beépített RAM memóriája 512 kbyte. A képgeneráláshoz CGA és MDA kártyát egyaránt beépítettek. Lényeges, hogy beépített UHF TV modulátorral rendelkezik. A panel 8087 típusú matematikai co-processzorral bővíthető, valamint ROM BIOS-t is tartalmaz.

A gép ajánlott operációs rendszere az MS-DOS 3.3, amely a géphez tartozó mágneslemezen is megtalálható. Az alapegység jobb oldalán elhelyeztek egy 3 1/2"-os lemez-meghajtó egységet, amely 720K kapacitású, de természetesen a géphez csatlakoztatható külső 3 1/2" ill 5

1/4" méretű lemez-meghajtó egység is, hogy könnyű legyen adataink átvitele az egyik formátumról a másikra. A gép hátoldalán több szabványos port-ot helyeztek el. A legnagyobb módú megoldás az IBM gépekhez hasonló bővítési lehetőség. Utólag 2 bővítő kártya behelyezése nyílik lehetőségünk, az egyik leghasznosabb bővítés a winchester típusú tárolóval való kiegészítés, a géphez kártya + tároló kb. 300 angol Fontért már be is szerezhető. Beépítettek szabványos RS-232 és CENTRONICS port-ot, valamint találunk a gépen analóg joystick port-ot is.

A hálózati tápegység is a gépben kapott helyet, úgy mint a beépített hangszóró, amelynek szabályozható a hangere ssége.

Azt gondoljuk nem kell különösebb kommentárt fűz nünk a gép software ellátottságához, hiszen az IBM PC XT/AT kompatibilis és 512 kbyte memórián belül MS-DOS alatt futó programok kivétel nélkül futtathatók ezen a gépen is, így nem kell megjedni, ezt a SINCLAIR gépet sem hagyják cserben a játékok, az utóbbi időben még nagyobb százalékban nőtt az IBM gépekre készült játékok száma.

A géphez monokróm vagy színes monitor illetve televízió készülék egyaránt csatlakoztatható. Mint tudjuk az IBM PC gépek legelterjedtebb grafikus kijelző szabványa a CGA rendszer. Ez lehetővé teszi, hogy 25 sorban és 40 oszlopban vagy 25 sorban és 80 oszlopban kezeljük a képernyőt 16 színű üzemmódban, karakterenként 8\*8 ponttal. Nagyfelbontású grafika 4 színnel 320\*200 képpontból építhető fel, ill. 2 szín felhasználásával 640\*200 képpontból.

Az új SINCLAIR gép kezelni tudja az MDA rendszert is (úgy mint az APPLE Macintosh gépek) monokróm üzemmódban, ekkor a 80\*25 karakteres képernyőt karakterenként 9\*14 pontból építi fel. A TV modulátoron keresztül természetesen csak a CGA módban használhatjuk a gépet.

A SET olcsónak számít a PC kategóriában, ha az alapkonfigurációt vesszük meg, ez áll magából az alapgépből (a gépet az otthoni TV-re kötjük), tartozik hozzá egér, software-ek: MS-DOS 3.3, GW BASIC, GEM-3 Desktop; valamint egy részletes használati útmutató. Mindezt együtt 299,- angol font-ért vesztegetik. Ha kérünk hozzá joystick-et, monokróm monitort, egy 'ORGANIZER' nevű programcsomagot, valamint 4 játékprogramot, ez esetben már 399,- fontot kell fizetnünk, s végül amennyiben a monokróm monitor helyett színeset szeretnénk, úgy 499,- font a SET ára.

A gép joggal nevezhető a PC-k egy felsőbb lépcsőfokának, súlya mindössze 5,4 kg., s mivel hordozható, valószínűleg sikeres lesz a munkahelyeken, iskolákban egyaránt.



**DOUBLE DRAGON - Melbourne House**

Ez a program is egy eredeti arcade játék kitűnő átirata. A történet röviden a következő: barátainkat a szemünk láttára elrabolja két bandita; mi a teendő? Elszaladni sajnos nem tudunk (abba az irányba a képernyő nem hajlandó scroll-ozni), így nem marad más hátra, mint a bátrabb út választása. A banditák után eredünk, hogy kedvesünket kiszabadítsuk. Amint az várható, természetesen mindenki megpróbál minket feltartóztatni. Nincs más kiút, fel kell venni a küzdelmet a minden irányból támadó ellenségekkel szemben. A mozgatható tárgyak - baseball ütő, korbács, kés, doboz, kövek - fegyvernek számítanak ellenségeink kezében éppúgy, mint a mienkben. Természetesen szükség van a fegyver nélküli küzdelemben való jártasságra is: ököl, könyök, lábfej, térd használata nemcsak megengedett, hanem szükséges is. Ha átverekedtük magunkat a három nagy szinten (a 128K-s verzióban még több van), a nagyfőnökkel, Big Willy-vel kell szembeszállnunk (neki sajnos géppisztolya van). A küzdelmi játékok kedvelői számára a **Double Dragon** igazi csemege. A grafika remek, a játékok egy és két játékos is játszhatja. Némiképp hasonlít a **Target Renegade**-c: játéka, de talán még nehezebb.

**ROAD BLASTERS - US Gold**

Az **US Gold** cég azok közé a software forgalmazók közé tartozik, amelyek holmi saláták megjelentetésére nem vállalkoznak. Játékaik - tisztelet a kivételnek - színvonalasak, élvezet a velük való játék. A cég új játéka autóverseny a javából. A gép - kétajtós **GWB 68 Turbo** - ígéretek: kb. 350 km/óra (nitro-befecskendezéssel 480 km/óra) csússsebességre képes. Mindez jól hangzik, de a dolgok nem ilyen egyszerűek. Az országon át vezető úton mozgó (autók, dzsipek, motorkerékpárok) és álló (aknák, ágyúk, vegyi fegyverek) ellenségek próbálják megakadályozni a célbajutást. Az üzemanyag hamar elfogy, az ellenőrzési pontokon tölthető fel újra, továbbá útközben is: a zöld és piros foltok extra üzemanyagot és löszert jelentenek. Igen, löszert, mert az autó nem védtelen: állandó fegyverzete, a géppuska mellé különleges eszközök is beszerezhetők (cirkálórakéta, nitro-fecskendő, gyorstüzelő gépágyú, pajzs). A játék 50 állomása több menetben tölthető be. Az autó grafikája jónak mondható, a scrollozó környezet kissé monoton. Az út simán, a szinproblémákat elkerülve fut alattunk. Az útközések észlelése néha késik. Összességében: legalább olyan színvonalú, mint az **Out Run**!

**INTENSITY - Firebird**

Egy földi kolóniát idegenek támadtak meg, az otlévkők úgy döntöttek, hogy műszaki problémák miatt nem szállnak úrhajójukba. Megmenekülésük egyetlen lehetősége egy robotrepülő, amit mi irányítunk az úrcirkálóból. A kolónia tagjai egy napon kibújnak rejtekhelyükről, és feladatunk, hogy a robotrepülővel összeszedjük őket. Ezután ügyelnünk kell arra, hogy a repülő ne ütközzön a falaknak vagy az idegeneknek. A kezdetben ártalmatlannak látszó ellenség később óriási károkat képes okozni, ezért az úrcirkálóval mielőbb támadásba kell lendülnünk. Ha a repülővel elég embert sikerült megmentenünk, a következő képernyő vár. A játéktér öt - különböző nehézségű - szintre oszlik, mindegyik 16 képernyőből áll. Az utolsó képernyő jelenti a szabadulást. Az egyes szintek között is lehetséges átmenet, így a befejezéshez sok-sok út vezet. Mindez bonyolultan hangzik (meg sem említettünk minden lehetőséget), az **Intensity** mégis kitűnően játszható. A Firebird nem véletlenül lép olyan sikerrel a piacra, amely méltán pályázik az 'év játéka' címre: a program készítőinek előző alkotása az **URIDIUM** volt, amely a **Hewson Consultants** égizse alatt jelent meg.

**CYBERNOID II. (The Revenge) - Hewson**

Az, hogy az **URIDIUM** szerzői most a Firebird céget választották, koránt sem jelenti a **Hewson** hanyatlását. Amikor azt hittük, hogy egy játék, a **Cybernoid** elérte a 48K-ba bezsűfoltható játékok csúcát, rá kellett jönnünk, hogy tévedtünk. Az **Exolon** szerzője (Raffaele Cecco) 'rátett még egy lapáttal'. A **Cybernoid II.** nehezebb is, kidolgozottabb is elődjénél. **Cybernoid**-unk most is a megszokott ötféle fegyverrel rendelkezik, az úton azonban felvehetők és az ellenség megsemmisítésekor szerezhetők további fegyverek. Ezek közül kétség kívül leghasznosabb a **CYBERMACE**: környezetünkben mindent szilánkokra robbant. Persze ellenségeink is sokat fejlődtek. Most minden irányból támadnak, emellett néhány fegyver elhelyezése ellenségeket álcáz, amelyek éppen akkor rontanak ránk, amikor egy jó találatnak örvendezünk. Nem tanácsos egy képernyőn sokáig időzni - vigyázat, a csaletek halálos csapdát rejt. A lefelé vezető járatokon is nehezebb átjutni. Ismerjük a **Hewson** korábbi játékeit, így méltatásul csak annyit: a **Cybernoid II.** minden szempontból előrelépés, s hogy Raffaele tud-e még valamit tőnni a memóriába? Várjuk a folytatást!

**Out Run**

Az idő megállítását a főprogram 111/5864/40960 file-térképpel rendelkező verziójához szemléltetjük.

Töltsük be a loader-t, majd RESET, és írjuk be a következőt:

10 CLEAR 24575: LOAD "" CODE: RANDOMIZE USR 28350: LOAD "" CODE: POKE 40623,0:  
RANDOMIZE USR 31051

Futtassuk az új loader-t, majd indítsuk a magnetofont.

**Hysteria**

A sérthetatlenség beviteléhez a BASIC/6912/40192 file-térképpel rendelkező verziót választottuk ki.

Töltsük be a loader-t, majd RESET, és írjuk be a következő sort:

10 CLEAR VAL "24999": LOAD "" SCREEN\$: POKE 23739,111: LOAD "" CODE: POKE 44588,201:  
RANDOMIZE USR 45066

RUN, majd a SCREEN\$ blokk-tól töltsük tovább a programot.



2499-ben a túlnépesedett földet éhínség fenyegeti, az emberek egymás ellen harcolnak az élelemért. A Föld urai az emberiség történelmének egyik legnagyobb tervét dolgozzák ki: kezdetét veszi egy hatalmas űrállomás építése, amelyen ezer és ezer ember fogja gyors ütemben termelni az élelmet. Az űrállomás neve: **NORTHSTAR** (Sarkcsillag).

A terv a sikeres befejezéshez közeledik. Élénk a forgalom az állomás és a Föld között. Egy napon azonban az egyik szállítmány eltűnik. Az űrállomás nem válaszol a Föld hívására. Nekünk kell elrepülnünk a **NORTHSTAR**-ra, és ki kell derítenünk, hogy mi történt.

Amikor megérkezünk, azt látjuk, hogy az állomást idegenek foglalták el. Emberi túlélőnek semmi jele: vajon mindannyian meghaltak? A létfenntartó berendezés nem működik, mi is csak azért maradunk életben, mert félig robotok vagyunk. Feladatunk egyértelmű: el kell pusztítani az idegeneket, meg kell találni és üzembe kell helyezni a létfenntartó berendezést.

A játékban folyamatosan látható a pontszám, az oxigénszint és az összegyűjtött fegyverzet. Kezdetben csupán robotkarjainkkal küzdhetünk, menet közben pótolható azonban az oxigén és további öt fegyver vehető fel. Ezek használata nyilvánvaló, csak a bomba igényel külön billentyűt. Egy élet elvesztésekor egy fegyver is elvész, de újra begyűjthető.

A játék irányítható **KEMPSTON, CURSOR III. INTERFACE II.** joystick illesztőkkel, valamint billentyűzetről is, ami átdefiníálható. Az eredeti billentyű-kiosztás:

K - bal  
SPACE - tűz

L - jobb  
S - bomba

A - fel  
Q - vége

Z - le  
P - szünet

A játékban több **POKE** segítségével eredményesebbek lehetünk:

POKE 44433,0 - nincsenek idegenek

POKE 48371,36 - végtelen élet

A következő betöltő segítségével is aktivizálhatjuk az örökleletet. A nagy kód elé álljunk a magnetofonnal, majd az itt látható **BASIC** sorok futtatását követően indítsuk el a magnetofont, s betöltést követően végtelen életünk lesz:

10 BORDER 0: PAPER 0: INK 0: CLS

20 LOAD"" CODE

30 FOR f = 65448 TO 65455

40 READ a: POKE f,a: NEXT f

50 RANDOMIZE USR 65428

60 DATA 62,36,50,243,188,195,191,183

A játék egyes szintjeit (1-10) a mellékelt térképen nyomon követhetjük. Az egyes objektumok jelölései:



- BONUS

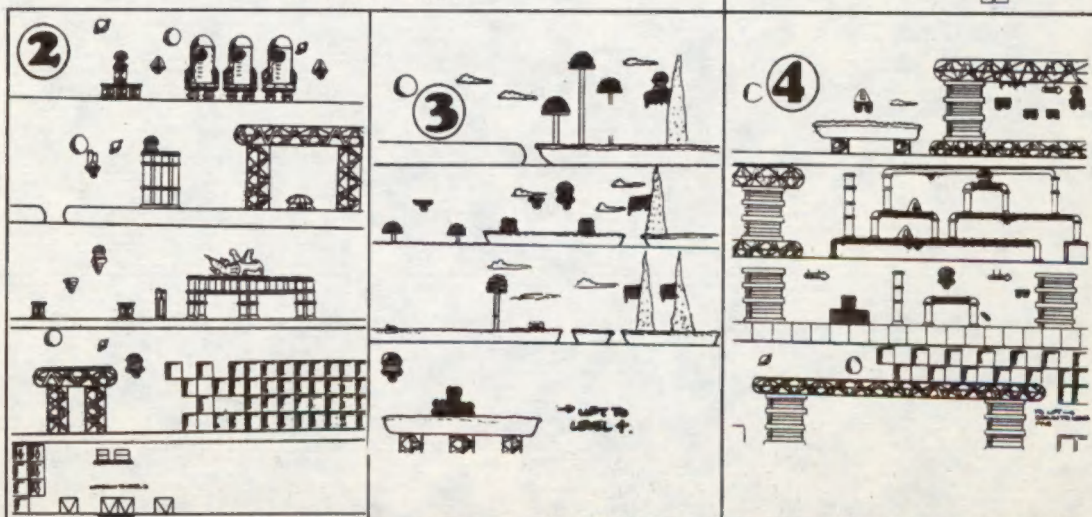


- Fegyverzet

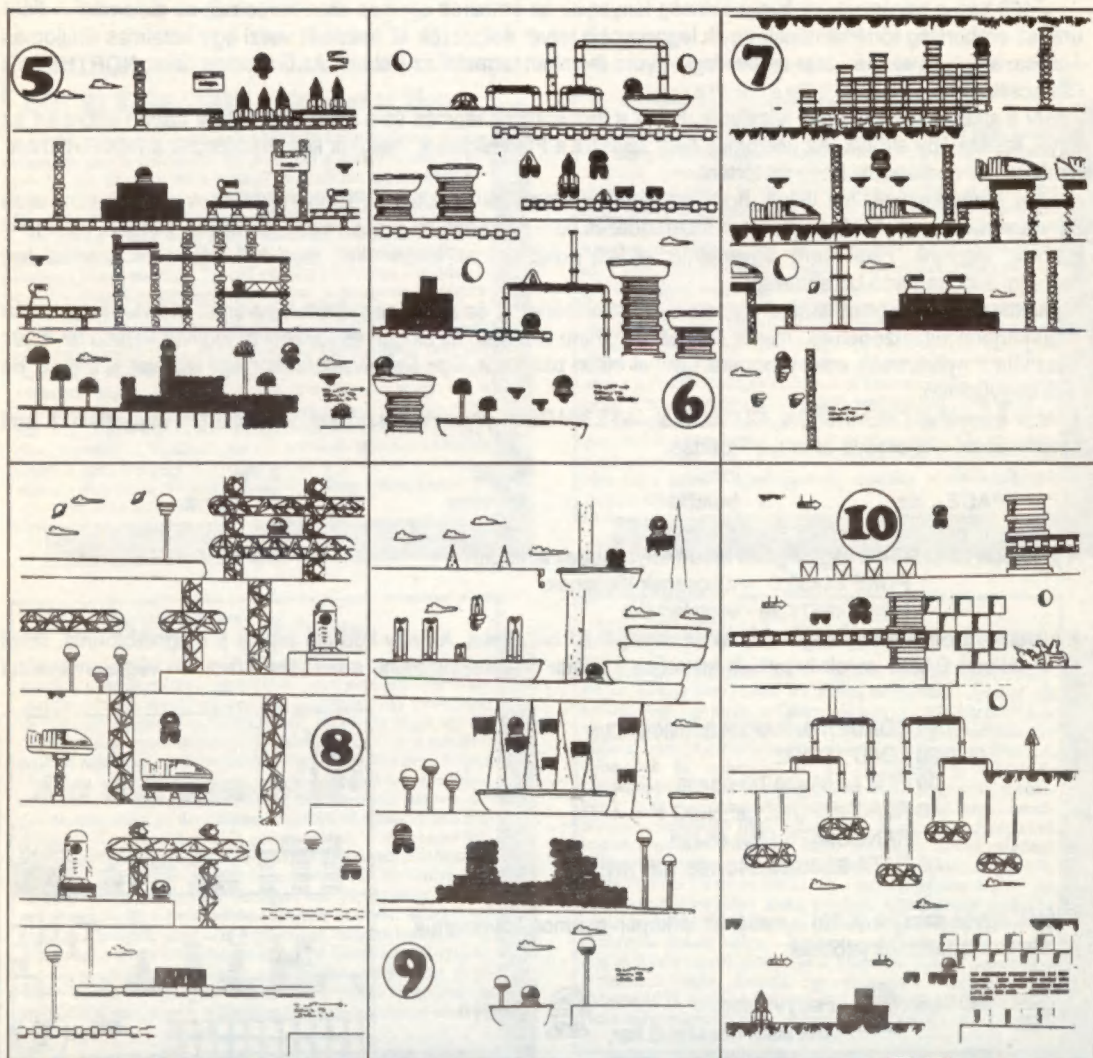
(Az első: hosszabb kar,  
a második: kar felfelé,  
a harmadik: bomba)



- Oxigén







## Game Over I.

Az örökélet bevitelét a BASIC/6912/33833 file-térképpel rendelkező verzióra oldottuk meg.

Töltsük be a BASIC loader-t, majd állítsuk le a magnót. RESET, és írjuk be az új loader-t:  
 10 CLEAR 65535: LOAD"" CODE: FOR i=25037 TO 25046: READ a: POKE i,a: NEXT i: RANDOMIZE  
 USR 25000

20 DATA 175,50,173,152,50,166,153,195,132,123

Futtassuk a programot (RUN), és indítsuk tovább a magnót.

## Game Over II.

Az örökélet és indítási kód bevitelét a BASIC/6912/33833 file-térképpel rendelkező verzióra végeztük el.

Az örökélethez a 38398 és 38692-es címekre kell zérust elhelyezni. Először is töltsük be a loader-t, és állítsuk le a magnót. RESET, majd írjuk be a következő programot:

10 CLEAR 65535: LOAD"" CODE: FOR i=25037 TO 25046: READ a: POKE i,a: NEXT i: RANDOMIZE  
 USR 25000

20 DATA 175,50,254,149,50,36,151,195,132,123

Futtassuk a programot (RUN), és indítsuk a magnetofont.

Betöltés után a 4. opciót választva írjuk be a következő kódot: 18024, ezután már indítható a játék.



**T**íz éve született **Garfield**, a tohonya, kövér és más hasonló érdemekkel felruházott, de mégis szeretetre méltó macskaféle. Ez alkalomból jelent meg, a rajzfilmsorozat számítógépes változata. **Garfield** egy hosszú komikus történet főhőseként született, de a rohamosan növekvő **Garfield**-mánia révén, hamarosan feltűnt a pólókon, naptárokon és a képregények lapjain. Az átlagember-legalábbis az angol átlagember – ismeri **Garfield**-et, aki egyike ama figuráknak, akit az 5-95 év közé eső korosztály, egyaránt figyelemmel kísér.

Tim Langdellnek, a játék írójának, bizony sok gondja akadt, mikor egy computerre átdolgozható sztorit keresett. A rajzfilm alkotói ugyanis kiűzték minden olyat, ami az eddigi comics sorozatok szinte mindegyikére jellemző volt; azaz halált a gonoszt és a többi hasonló ponyva kelléket. **Garfield** nem káromkodhat, nem beszélhet közönségesen, nem ölhet, sőt nem is tehet tönkre senkit. Nekí csupán egy speciális erkölcsi morálja van, és valószínű ez az ami népszerűségét biztosítja. Ő csupán eszik, iszik, néha elereszt egy-egy szitkot, utálja a hétfőt, és gyakran esik álomba. (Mindezek a tulajdonságok, szinte a megszólalásig megegyeznek a *Piranha* – *Yogi Bear* című játékanak főhősével. Ebben a főhős – Maci Laci – ugyancsak sokat alszik, lopja a túristáktól a piknik kosarakat, de kárt nem tesz senkiben.)

A gyakorlatban a legmegfelelőbb epizódok a *Big, fat, hairy deal* című rész tűnt, melyben **Garfield** agresszív tulajdonságai fokozottabban jönnek elő. A programozók lényegében az epizód teljes feldolgozását tervezték, méghozzá egy 16 bites computerre. Csupán próbaként írták meg a játékot **SPECTRUM**-ra, de az olyan jól sikerült, hogy Tim Langdell és csapata itt megállapodott. A játékírók már biztos fogásnak látták **Garfield**-et, aki teljesen belemaradt a Spectrum monitorok és Ulák szívébe. Az *Edge* a hatalmas sikerre való tekintettel, a rajzfilmhez hasonlóan, egy marathoni hosszúságú játéksorozatot tervez, aminek a komikus epizódok egy-egy részét dolgozni fel. Ehhez a munkához, Jim Davis, már korábban is készített egy macska kreátor nevű, kifejezetten **Garfield**-ek készítésére alkalmas segédprogramot. Valószínűleg, a *Big, fat, hairy deal* egy nagy görgeteg elindítója, így a következő rész – már jövő áprilisban elkészülhet. **Garfield** népszerűsége talán a computereken is töretlen marad, tehát ne hagyjuk őt az autóban (poén egyébként, hogy a játék megírását az illető, hogy Angliában egyetlen week-end alatt 45 autót tört össze, és amelyekben volt **Garfield** figura – plasztik bábú formájában – azt egytől egyig ellopták).

A *Big, fat, hairy* kivitelezése ragyogó, az ember teljesen bele tud merülni a játékba. A tökéletes rajzfilmkarakterek, azok animációja, a Spectrum élesvonalú grafikái, és az ehhez párosuló frappáns árnyékolások, igen látványossá teszik a programot. Több ötlet, meglepetések kerülnek elő szinte minden "sarokból". A játék teljesen a rajzfilm stílusát követi. Egyedül a hang és a szín az, amit hiányolhatunk. Már egy jó *Jack the Nipper* jellegű dallam is feldobta volna a programot (sajnos 128K-s zene sincs – legalábbis nekünk). A legelragadóbb mégis a kombinációs készség. Játshatjuk a játékot kalandszerűen (azt vigyázz, azt tegyük le oda stb.). A könnyű szórakozásra vágyók azonban mehetnek csupán szavakra és pontra, anélkül, hogy végig kellene játszaniuk a játékot.

A történet: **Garfield**, a kövér, kivárgzott, folyóvíz-géregűz, kedd lévén úgy döntött, hogy valami hasznosat csinál. Kis macskabarátját – illetve barátját – **Arléne**-t, ugyanis megfogta a kéményseprő egy éjszakai macskakonzert közben, amikor a C-korolyka duót adták elő, 5 KHz-en. Ez kellemetlenül érintette a közeli lakók és a kéményseprő dobhártyáját. Így, **Arléne** hamarosan a hatóságilag kirendelt gyepmester, városi végrehajtótelepére került. **Garfield** feladata megkísérelni **Arléne** kiszabadítását a sintér karmai közül.

Hősünk, tizedik születésnapjára nem kért mást, mint kaját és hogy hagyják békén pihenni. Macskánkánál tehát mindig kell lennie valamirevaló elemőzsiának, mert ennek folyamatos adagolása nélkül nagyon elfárad, elmegy aludni, és **Arléne**-t a sintér karmai között hagyja sorvadni. De ne féljünk, hogy **Garfield** nagyon inycen lenne, és esetleg éhségstrájkba kezd. Táplálékait a következő intervallumon belül számíthatjuk: minimum csont, vagy száritott hús, felső határ nincs. Mint később látni fogjuk ez valóban így van, hiszen **Garfield** a hentesből lopott kolbásztól, **Odie** (**Garfield** kutyabarátja) csontjáig, szinte mindent képes felfalni. Még **Jon** (**Garfield** mestere; azaz a gazdi) kávéját is megpróbálja elemelni, ez azonban nem sikerül, mert ilyenkor **Jon** keményen a szemébe néz, és hősünk kénytelen elsomfordálni.

**Garfield**-et gyakran akadályozza, vagy segíti **Odie**, egy kajtató fajtájú eb. **Odie**, igen ragaszkodó természetű és kilógó nyelvvel, szinte mindenhová követ minket. **Odie** keserűségére, **Garfield** nem mindig viszonyozza dogi (=kutyai) érzelmeit, és egy jól irányzott fenékkerügással elküldi a francba. **Odie**, gyakori memóriakiesése folytán, lankadatlan hűséggel ugrál vissza hozzánk, újabb fenékkerügásokért. **Garfield**-nek azonban van más feladata is... Segítenie kell ifjú unokaöccsén,



**Nermál-on**, aki egy felhúzhatós egér üldözése közben a város földalatti csatornarendszerének egyik sötét pontjára került. Igaz, az egeret elkapta - mivel az lejárt - de a sötétből kifolyólag, kijönni már nem tudott.

**A játék megkezdése előtt nézzük át az irányítást:**

- 1 KEYBOARD (Q,A,O,P,M,ENTER)
- 2 SINCLAIR
- 3 KEMPSTON
- 4 REDEFINE(FIRE,UP,DOWN,LEFT,RIGHT,PAUSE)

lövés=rugás    lövés és fel=ugrás    lövés és le=falás    le=felvétel

**Az irányítás után nézzük meg, mit is láthatunk a képernyőn:**

Játék közben, beleláthatunk **Garfield** gondolataiba. A screen 'lábánál' megjelenő mondatok, kellőképpen tolmácsolják éppen aktuális érzelmeit. A szöveg fölött látjuk, hogy mi van nálunk. Itt maximum három tárgy lehet. A legfelső van **Garfield**-nél, a középső **Odie**-nél, míg a legalsó **Nermál**-nál. Ebből következik, hogy **Odie** és **Nermál**, szubjektív belátása szerint vesz fel, vagy tesz le dolgokat, gyakran éppen **Garfield** orra előtt ilyenkor jogosan éljük ki agresszív hajlamainkat, vagyis a nekünk kellő tárgyakat néhány fonékberúgás segítségével kobozzuk el a két 'úriembertől.'

Ezek után, már tényleg el kell kezdeni a leírásunkat, mert **Arléne** trotyli vénmacska lesz, mire odaérünk. Első, egyben legnehezebb feladatunk a sintértelep kulcsának a megszerzése lesz. Ehhez **Odie**-ra is szükségünk van, habár amúgy is utánunk koslatna (nyelvét bal jobb irányba lóbalva). **Garfield** terve, felkeresni óriáspatkány barátját. **Csatornai Futkosit**, akiről köztudott, hogy minden vacott begyűjt és hamarosan fel is fűt. Étrendjén a legkülönbözőbb ételek, (mint pl.rendszám tábla faforgáccsal, ujsággal, szőráknyerrel) szerepelnek. Sietnünk kell mert **Garfield** egyik egérbarátja mondta, hogy **Futkos** hozzájutott a városi sintértelep kulcsához, és egy **Odie**-től kapott csont kíséretében, hamarosan vacsorára idehaza elfogyasztani őket. Eledeleit mindenki elől elzárva, egy nagy ládában tartja.

Induljunk tehát **Odie** kíséretében az óriáspatkányhoz (a térképen ld. patkány és a láda). A láda kinyitásához "macsuka kirikiri" nevű, japán karate rugásunkat kell használni. Van azonban egy gond. Rugásunk ugyan kinyitja a ládát, de **Futkos** rögtön lecsap tartalmára, s a legbiztosabb helyre, saját gyomrába rejti (itt kell megemlítenünk, hogy patkány barátunk, bármilyen orra elé rakott tárgyat örökre eltüntet, így itt ne nagyon csináljunk kereskedelmi csomópontot). A patkány étvágya, **Garfield**-et fokozott félelemmel tölti el és ha sokáig itt tétovázunk, félelmében mély álomba esik, s a játéknak vége szakad. **Odie** azonban nem így reagál. Nyugodtan ugrál a sötét pincében, anélkül, hogy bármilyen baja érne. Ez lesz a mi segítségünkre. Álljunk be közvetlenül a láda mellé (ld. a képen), majd mikor **Futkos** távozik, **Odie** viszont a doboz közvetlen közelében van, rugjunk bele a ládába, és gyorsan vegyük fel a tartalmát. Ha ügyesek voltunk (ez a tizedik próbálgatás után sem valószínű) hozzánk került a csont, vagy a kulcs. A kulcs mindenképpen fontosabb, ezért ha **Odie**-nál lenne, a már ismertetett módon vegyük el tőle. Ha meg van a kulcs, már csak el kell jutnunk a sintértelepre. Ez igen bonyolult folyamat, hiszen a sintértelep a külvárosban van, jó messzire a citytől. **Garfield** szerencsére protekciót élvez. Unokaöccse ugyanis, egyszer megmentett egy fészekből kiesett madárlókát (felhúzhatós egernek nézte, és úgy megkergette, hogy a madár kénytelen volt időnk előtt a repülés művészetét elsajátítani). Madarunk azóta felnőtt sirállyá lett, s elelemgyűjtő portyái során gyakran száll a külvárosban található városigazgatási hulladékgyűjtő telep körzetébe. A minimális utiköltség fejében, hajlandó elszállítani (minimális utiköltség=madáreledel). Ennek megszerzése azonban egy macska számára igen nehézkes, mivel a boltokban csak humán lényeket szolgálnak ki. Ennek ellenére bizonyos trükkök segítségével sikerre számíthatunk.

Először is a fészkerbe kell elmennünk, a lapátért. Mellékesen a kulcsot rakjuk le, mivel itt biztos helyen van. Az ásóval induljunk el **Interface Iván** Hardware szaküzletébe. Belépve álljunk **Interface Iván** pultja alá, és csináljunk néhány faló mozdulatot (lövés plusz le).....**Interface Iván** elképed, hiszen még soha nem látott ást nyelegető macskát, ráadásul saját, hardware eszközöket árusító boltjában. Nagy megdöbbenésben, egy 1 fontos bankjegyet húz elő pénztárcájából, s **Garfield** lábai elé dobja (mondván, hátha ezt is megeszi). Ez az öröme azonban már nem lesz meg kedves barátunknak. További akrobatamutatványok helyett vegyük fel a bankjegyet, s nyargaljunk kifelé. A megszerzett pénzzel, most már nyílegyenesen tarthatunk a Tápszerek boltjába. Itt minden lény beszerezheti korszerű táplálkozási étrendjéhez a legmegfelelőbb ételeket. Belépve a szaküzletbe, energikusan adjuk le 1 fontosunkat. A szolgálatkész eladó látván **Garfield** méreteit, rögtön egy kalóriaszegény macskaeledelt traktál ránk. Sok választási lehetőségünk nincs, ezért kénytelenek vagyunk megvenni. Ezután menjünk az eledellel egészen a Parkig. Belépve, a parkőr bizarr



mosolya fogad, de ez ne sokat zavarjon. Tegyük le a macskaeledelt a pázsitra. Sirálybarátunk, ekkor nyomban megjelenik, de sas szemével rögtön kiszúrja, hogy ez nem neki való táplálék, s szépen tovaszáll. Mi is hagyjuk el a terepet (az eledelt is), hiszen még vissza kell mennünk a kulcsért a fészerbe. Visszatérve tegyük le a kulcsot, vegyük fel a madáreledelt, majd tegyük le a madáreledelt és ismét vegyük fel a kulcsot. Ekkor ismét megjelenik sirálybarátunk, azt gondolván, hogy tettünk egy fordulót a tápszerboltban, madáreledele érdekében. Pedig rosszul hiszi...Ezt ő is tudja, s ismét tovaszáll. Persze azt is rosszul hiszi, hogy ilyen könnyen lerázhat bennünket. Egy Garfield-től hihetetlen nagyságú ugrással lendülünk a madár karmai közé. Ő a gravitációra fogván a dolgot, kényelmesen száll tovább, egészen a külvárosig. Itt hősünk landolókészülékai automatikusan kioldanak, s már csak annyi a teendők, hogy bemegyünk az ajtón és a kulccsal elmegyünk jobbra.

Arléne boldogan szalad Garfield felé, a cél sikerült. Arléne ugyan szabad, viszont a 100% elérésétől még messze vagyunk, hiszen mint már említettük, teljesen végig kell játszsanunk a filmbeli történetet. Még igen sok kaland és kellemes meglepetés vár ránk. A továbbiak kifejtését Önökre bizzuk, s jó szórakozást kívánunk.

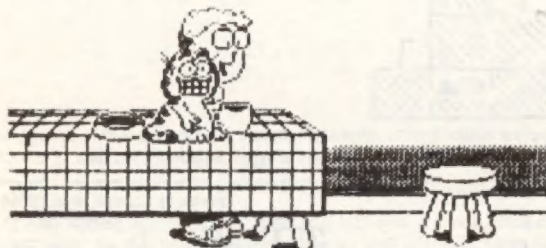


Koszor patkány, megszerzem a kulcsot!



Már megint itt van ez a hülye kutya!

Pfű, de sötét van odalent!



Hagyj békén, Garfield!



Takarodj, te kövér macska!

## Renegade

Az örökélet beviteléhez a BASIC/33/6912/40704/1410 file-térképes verzió mellett döntöttünk. Töltsük be a BASIC loader-t, majd RESET, és írjuk be a következőket:

```
10 CLEAR 65000
20 FOR i=65001 TO 65037: READ a: POKE i,a: NEXT i
30 DATA 62,221,33,0,91,17,0,159,55,62,255,205,86,5,49,255,255,62,195,50,88,160,195,203,92,50,72,92,
  205,107,13,33,0,61,17,48,0
40 LOAD ""SCREEN$: RANDOMIZE USR 65001
```

A betöltést RUN-t követően folytassuk a SCREEN\$ file-tól.

## Z

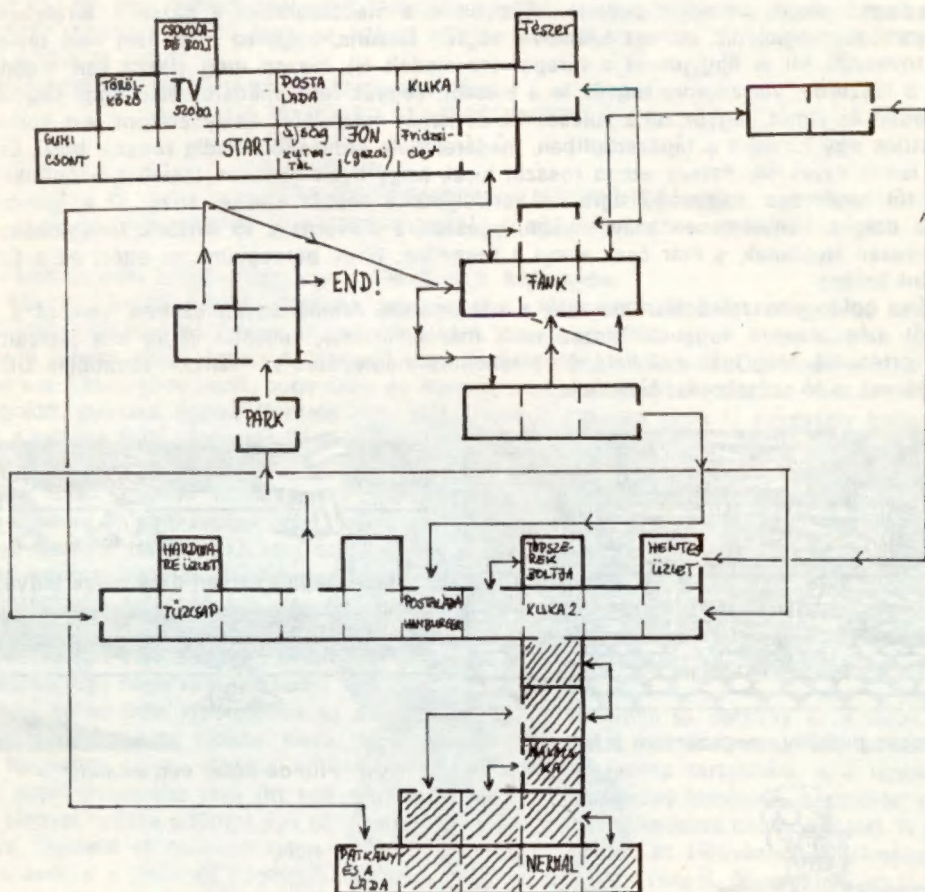
Az örökélet beviteléhez a BASIC/6912/41072 file-térképpel ellátott verzióra érvényes.

Töltsük be a loader-t, RESET-eljünk, majd írjuk be a következő programot:

```
10 CLEAR 24319: LOAD "" SCREEN$: LOAD "" CODE: POKE 41285,0: RANDOMIZE USR 41072
```

Futtassuk a programot (RUN), és indítsuk el a magnetofont.





## ZORRO

A nő kendőjét szerezzük meg a kútról / Menjünk le a kútba és keljünk át a vízen / Menjünk le a virágért, és ezt vigyük fel a labdához / A labdát lökjük le, és álljunk rá a virággal együtt a visszaérkező liftre / Menjünk vissza a vízhez, és tegyük le a virágot / Keljünk át a vízen, és másszunk át a kúthoz / Szerezzük meg a kulcsot, és nyissuk ki az üveghez vezető ajtót / Szerezzük meg az üveget, és itassuk le az őrt / Az őt hasán ugrálva jussunk fel az emeletre, és lökjünk le egy őrt a kampóra / A csomag alatti lépcsőn menjünk le; és szerezzük meg a poharat / Szerezzük meg a 'Z' alakú tárgyat (billo), majd másszunk fel a fára, és ugorjunk át a falra / A 'billo'-ot melegítsük fel, és bélyegezzük meg a bikát / Szerezzük meg a bika melletti villogó tárgyat / Szerezzük meg a trombitát, majd álljunk rá a billegő ugratóra, és fújjuk meg a trombitát / Menjünk fel a csizmáért / Szerezzük meg a két kis harangot, és tegyük le ezeket a temetőben található harangtoronyban / Másszunk le a sírba, és keressük meg a pénzt (\$) / A villogó tárgy által jelzett kijáraton át jöjünk ki a várba / Nyissuk ki az összes ajtót (vigyázzunk, közben lőnek) ? Menjünk tovább a következő képernyőn / Másszunk fel a rabnőhöz / Szerezzük meg a fekete rőzsát / Ezt is vigyük el a síron, ill. az előbb leírt úton át a rabnőnek (100 % !!!)

## THING

Használjuk a telefonkönyvet a telefonfülkében a teleportáláshoz / Teleportáljunk 'T'-be / Balra / Vegyük fel a vörös mega-heringet (RED MEGA-HERING) / Használjuk kétszer / Teleportáljunk 'K'-ba / Jobbra / Vegyük föl a kalapot (HAT) / Használjuk kétszer / Teleportáljunk 'J'-be / Jobbra / Vegyük fel az AMERICAN EXPRESS kártyát és használjuk azt / Teleportáljunk H-ba / Balra / Vegyük fel a Lobotomy készletet (LOBOTOMY KIT), majd használjuk azt / Teleportáljunk 'L'-be / Balra / Vegyük fel a konzervnyitót (CAN OPENER) / Menjünk 'C'-be / Balra / Vegyük fel a konzerv ételt (CAN OF FOOD) / Használjuk a konzervnyitót (CAN OPENER) / Teleportáljunk 'Y'-ba / Jobbra / Vegyük fel a tejeseveget (BOTTLE OF MILK) / Teleportáljunk 'B'-be / Jobbra / Vegyük fel a kupaknyitót (BOTTLE OPENER), majd használjuk azt / Teleportáljunk 'A'-ba / Jobbra / Vegyük fel a piszkos hamburgert (DINGABURGER) / Teleportáljunk 'W'-be / Balra / Vegyük fel az erős mustárt (CRAZEE MUSTARD), majd használjuk azt / Teleportáljunk 'F'-be / Balra / Vegyük fel a video-lejátszót (VIDEO PLAYER) / Teleportáljunk 'N'-be / Jobbra / Vegyük fel a BRUCE LEE video kazettát / Használjuk a video-lejátszót / Fájni kezd a fejk / Teleportáljunk 'S'-be / Jobbra / Vegyük fel a fejfájás elleni tablettát (HEADACHE TABLET), majd használjuk azt / Menjünk 'G'-be / Jobbra / Vegyük fel a VIRGIN T-SHIRT-t / Teleportáljunk 'Q'-ba / Balra / Vegyük fel Mates csomagját (PACKET OF MATES), majd használjuk ezt / Teleportáljunk 'E'-be / Balra / Találkozunk a nagy OVERLORD-dal (100 %)



Fíverünk Matt, miközben 'Mr. Big' sötét üzelmeit próbálta leleplezni, áldozatául a nagy gengszterfőnöknek. A 'szemet-szemért' elv szerint megpróbáljuk ezt megbosszulni: Scumvill városának öt kerületén átverekedve magunkat, hogy leszámoljunk a nagyfőnökkel.

A játékot egy és két játékos játszhatja. Két játékos esetében a ellenfelek is többen vannak. Ajánlatos együttműködni a külön-külön harc helyett.

A játék során a fegyver nélküli küzdelem szokásos elemei (ugrások, ütések, rúgások) használhatók. A fegyverrel támadó ellenségek legyőzése után ottmaradt fegyver felvehető. Az egyes szinteket adott időn belül kell teljesíteni, majd a kijáraton át távozva tovább lehet menni a következő szintre. A képernyőn folyamatosan látható a pontszám, az energia, a életek száma és a hátralévő idő is.

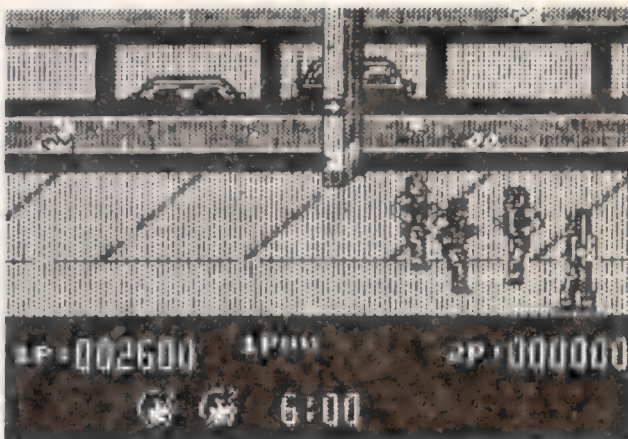
Az 1.szint a parkoló. Itt motoros gengszter támad, ugró rúgással kell a nyeregből kiutni. Ezután felkel és társaival együtt újra támad.

A 2.szint a sötét utca. Itt 'éjszakai pillangók' és főnökük az ellenfél. A főnöknek pisztolya van: addig löv, amíg a löszere el nem fogy, ezután belemegy a közelharcba.

A 3.szint a park. Itt néhány meglehetősen ellenszenves bőrhajút kell leküzdeni.

A 4.szint a bevásárlószertár. Itt nehézfiúk próbálják megakadályozni a továbbjutásunkat.

Az 5.szint a bár. Itt először a főnök testőreivel, majd magával Mr Big-gel kell szembeszállnunk.



A játék KEMPSTON III. SINCLAIR Interface mellett billentyűzetről is játszható, a átdefiníálható. Az eredeti kiosztás a következő:

	bal	jobb	fel	le	tűz	szünet
1.játékos	←	→	↑	↓	SP	P
2.játékos	6	7	↑	↓	0	P

Az akciók a következők (az alak jobbra néz):



Tűzgomb nélkül a megfelelő irányba haladunk. A fenti akciók iránya megfordul, ha a balra néz. A tűzgombnak egyébként több funkciója van: ütés, tőrdelés, fegyver felvétele és használata.

Az általunk javasolt stratégia:

Az első szinten ugró rúgással fogadjuk a motorost. A földre került ellenségeket célszerű megdölgözni, tapasztalatunk szerint azonban csak akkor, ha már másodszor fekszik le (az első után még túl frissen kel fel). Ha megérkezik a fegyveres, vegyük el tőle a tüőt minél előbb: nemcsak hatékonyabb, hanem több pontot is eredményez, mint a fegyver nélküli harc. A második szinten a 'hölgyek' következnek. Célszerű megvárni, amíg főnökük a tőlünk balra állót lelövi és elfogy a tölténye. A harmadik szinten mihamarabb érjük el a földön heverő fegyvert. Ennek segítségével gyorsan utat vághatunk a kijáratig. A negyedik szinten a játék már keményedik. A fiúk lehúzzák fejüket a jól bevált ugró rúgás elől. Érdemes tehát ezzel a mozdulattal átugrani felettük, majd két hátsó rúgással földre vinni őket. Johet a földharc. Az ötödik szinten a legeredményesebb az előbb begyakorolt technika gyors végrehajtása. A testőroket már az első letétkvör meg lehet dolgozni a padlón. Vegyük el a biliárddáktól és próbáljuk megtartani a főnökkel való találkozásig. A főnökkel való harc meglepő módon a játék egyik legkönnyebb része, ha felvesszük a padlóról a jól látható dáktól. Ha a főnök a közelünkbe ér, ussuk meg kétszer, ettől kikefekszik. Ezt négyszer megismételve győztünk. Matt békében nyugodhat.

A játék megoldását természetesen elősegítheti néhány POKE alkalmazása:

POKE 59911,0: POKE 63034,0 - végtelen élet  
POKE 62936,0 - 2 óra megáll  
POKE 63797,n - 2. játékos életeinek száma  
POKE 63760,n - idő (percben)

## Driller

Az örökélet beviteléhez a BASIC/6912/38125/6912 file-térképpel rendelkező verziót választottuk ki.

Töltsük be a loader-t, RESET, majd írjuk a következőket:

10 CLEAR 27469: LOAD ""CODE 16384: POKE 23739,111: LOAD ""CODE: POKE 47904,0: POKE 49021,0: POKE 49022,0: POKE 49024,0: POKE 49425,0: CLS: LOAD "" CODE 16384: RANDOMIZE USR 38820

Futtassuk az új loader-t, majd indítsuk tovább a magnetofont.



Cuchulainn, ■■■ ókori kelta hős épp kedvenc karosszékeben üldögélt és ■ **Spectrum Világ** legújabb számát lapozgatta, amikor hirtelen megszólalt ■ telefon. "Cuci, baj van, azonnal segítened kell..." - segített egy elhaló hang. "Mozdulj?" - kiabálta dühösen Cuchulainn, mert utálta, ha Cucinak nevezik. "Legjobb barátodat, Loeg-et elrabolta Skar, ■ huncut vécénén (másodállásban bozorkány), mert Loeg padlizsánnal dobálta őt köteleességének teljesítése közben. Most Darach városába hurcolták és ott elrejtve tartják fogva. Azonnal indulnod kell..." - és ■ készülékéből már csak egyenletes zúgás hallatszott (a telefonkábel a Keltáknál is be szokott ázni...). Cuchulainn értetlenül bámulta ■ telefonkagylót, majd hirtelen mozdulattal lecsapta. Gyorsan kiválasztotta garderobéjából a legszebb, krokodilfejes "lákost" pólt (a TIR ■ ■ ■ NOG óta kicsit elhízott, ergo most már nem kívánja a testét mutogatni ■ hölgypublicumnak) és elindult Darach városába, hogy megmentse barátját ■ gonosz ■ ■ ■ kezeli közül...

Valahogy ilyen előzményei lehetnek a Gargoyle cég **DUN DARACH** című arcade/adventure játékának (ebben azért nem vagyunk olyan biztosak), amelyben kedvenc hősnök újabb hihetetlen kalandokba bonyolódik egy elátkozott városban. A játék felépítésében, animációjában és izgalmában a **TIR NA NOG**-sikerreptől alapul. A kozlekedésre, mozgásra, irányító billentyűkre és ■ játékszabályokra vonatkozó instrukciók megegyeznek ■ **TIR ■ ■ ■ NOG** leírásánál megismertekkel (Spectrum Világ 8., akinek nincs ■ ■ szerkesztőség címén megrendelheti) - ezekre most nem kívánunk külön kitérni, csak ■ különbségeket fogjuk felsorolni.

Darach városa elátkozott hely, bár látványlag olyanak tűnik, mint más normális városok. Gazdasági életét ■ kereskedelem határozza meg, rengeteg árut adnak el illetve vásárolnak a boltokban, egyszerűen minden ■ "irid" néven emlegetett pénz körül forog. Természetesen ezzel ■ vonással együtt jár ■ bűnözés megjelenése is: ■ utcákon tolvajokkal találkozhatunk, sőt játékkaszinó is üzemel a városban. A kelta hősök fizetése nem túl magas, Cuchulainn mindössze 2000 iridvel ■ zsebében érkezik (igaz, hogy ez már adózás utáni állapot). Mint a későbbiekben kiderül, ez nem túl sok, legelső teendőnk, tehát az, hogy megnöveljük vagyonunkat. Ez többféleképpen lehetséges:

- Ha bekapcsolódunk a kereskedelembe, lassú, de biztos gyarapodásnak nézünk elébe. Mivel ■ ■ eladással foglalkozó kereskedők általában olcsóbban adják ■ árjukat, mint azok, akik megveszik, a do-log roppant egyszerű: pénzünkön árut vásárolunk, majd elvisszük egy olyan boltba, ahol ilyen árut vennének és felajánljuk megvételre. A vétel- és eladási ár közötti különbözet ■ mi hasznunk. ■ vásár-



lás a következőképpen történik: az 'ENTER' gombbal bemegyünk egy ajtón, amely mögött valamilyen árut árulnak (WE SELL...), megkeressük ■ elárusítópultot, levesszük az árut ■ asztalról ('Q'), majd az eladó előtt megnyomjuk a 'SPACE' gombot, amivel pénzt ajánlunk érte (IRIDI?). Ekkor ■ eladó elveszi tőlünk ■ árú ellenértékét és vihetjük a cuccot. Mivel ■ boltokban két asztal van, egy boltban egyszerre akár két adag árut is felvehetünk (összesen három tárgy, ■ ■ ■ árú lehet nálunk). Ezután elmegyünk egy boltba, ahol eladható ■ árú (WE BUY...) és ott ■ eladónak ■ 'SPACE' billentyűvel felajánljuk a kezünkben lévő (csillaggal jelölt) tárgyunkat (a csillagot a '8'/'9' gombokkal mozgathatjuk ■ nálunk lévő tárgyak között). Ha ■ boltos vevő az árura, rákérdez ■ nevére, majd pénzünket megnöveli ■ vételárral és udvariasan megköszöni fáradozásunkat.

- A kereskedelemnek egyszerűbb módja ■ lopás. Ez ugyan nem túl szép dolog, mindenesetre jövedelmező. A lopás egyszerűen történik: amikor elvettük ■ árut ■ asztalról, mellőzzük a fizetést és távozzunk ■ szájrával. Ilyenkor ■ elvett tárgy megnevezése mellett egy villogó "e" betűt láthatunk, ami jelzi, hogy ■ tárgy lopott (ez mindaddig ott is marad, amíg ■ ■ ■ fizetünk). A lopott dolgokat is eladhatjuk ■ kereskedőknek, viszont ■ utcán nem cserélhetjük el őket (ld. később Bren-nél). Leginkább aranyrudat (GOLDBARS) érdemes lopni. Ha lopott árut viszlunk, vigyázzunk nehegy találkozunk Ryde-dal, ■ önkéntes rendőrről, mert elkobozza és bírságként ■ összes nálunk lévő pénzt is elveszi.



- A meggazdagodás legegyszerűbb módja ■ szerencsejáték. Darach város egyik kerületében van egy játékbartlang, lomain Ludum, ahol nagy szerencsejáték folyik. Ez ■ szerencsejáték a kockázás, pénzben. Két asztalt láthatunk A és B felirattal, mellettük pedig két kijelzőt, amelyen két kocka forog. A kettő összegét kell eltalálnunk: ha ■ ■ ■ 5, 6, 8, ■ számok valamelyikére szavazunk, ■ B asztalra kell tennünk, ha ■ ■ ■ 2, 3, 4, 10, 11, 12 közül kérjük valamelyiket, akkor az A-ra. Ha a 7 jön ki a kockán, akkor ■ bank nyer (HOUSE WINS). Megfigyeléseink szerint általában a B ■ ■ nyer többször. A játékban nincs limit (legmagasabb tét), de minimum 200 iridit kell feltennünk. A "Tegyék meg tétjeiket!" (PLACE YOUR BETS) felszólítás után forduljunk a kiválasztott asztalhoz és ■ 'W' gomb nyomogatásával állítsuk be ■ tétet (egy megnyomás 200 iridivel emeli). A NO MORE BETS felirat után már nem emelhetünk, meg kell várnunk ■ következő dobást. Ha ■ kockákon az a szám látható, amelyikre tettünk, ■ tétben lévő összeg megduplázódik. A pénzünket a 'Q' használatával vehetjük ki (ha bent hagyjuk, akkor tétnek számít a következő dobásnál is). Mint már említettük, itt nagyon egyszerűen meggazdagodhatunk: felrakjuk az összes pénzünket, majd ha nyerünk, a '6' gomb megnyomásával visszatérünk a játék főmenüjébe és ■ második (SAVE GAME) opció használatával kimentjük ■ játékállást. Benthagyjuk ■ pénzünket ■ minden nyereség után kimentgetjük az állást. Ha veszünk, betöltjük ■ utolsó kimentett állást és próbálkozunk újra... Így elszórakozunk egy darabig, amíg össze nem szedünk kb. 30-40000 iridi-t (ha többet sikerül, ■ sem baj), ez körülbelül elég lesz ■ játék folyamán.

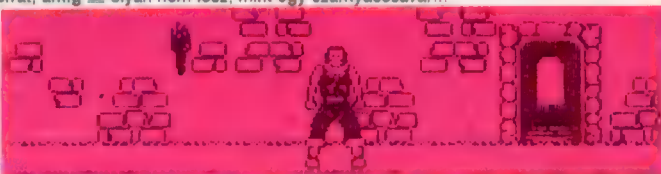


Ez tehát az első lépés. Ha összegyűjtöttünk egy nagyobb összeget, helyezük biztonságba a bankban, ahol egyrészt védve van a tolvajok ellen, illetve a kamatból kifolyólag újabb pénzeket fial számunkra. Az asztalnál a betétet a W billentyűvel tehetjük le, 1000 iridinként. Ezután a képernyő bal alsó részén külön kijelvezve lesz látható, hogy mennyi pénzünk van a bankban, illetve mennyi van nálunk (mindig csak annyi legyen, amennyi a következő feladat elvégzéséhez szükséges). Ezek a biztonsági intézkedések azért szükségesek, mert a városban nem csak mi szoktunk lopkodni, hanem két romlott hölgy (Kara és Keli) illetve egy úr (Kahn) is. Ezek elég helytelenül magukévá tették a tolvajtól lopni nem bűn című elgondolást és mesterségüket rajtuk gyakorolják: ha találkozunk valamelyikükkel szép jó estét (GOOD EVENING) kívánnak, aztán bocsánatért esedezve (YOUR PARDON) már el is lopták azt a tárgyat, amit a kezünkben tartunk (csillaggal jelölt tárgy). Ha nincs nálunk tárgy, sem baj, akkor pénzünket lopták el (űgyis általában az aktuális). Kivételek ez alól a kulcsként funkcionáló tárgyak, amelyeket nem tudnak ellopni tőlünk illetve a telestone. Mindenesetre legjobban ha óvakodunk a K-betűs emberekkel történő összeütközéstől.

Nem csak tolvajokkal, hanem más emberekkel is találkozhatunk (Mhor, Ryde, Pita stb.). Ha egy ember mozog azon a képernyőn, ahol mi is vagyunk, a jobb alsó részen megjelenik a neve is. Ha többen is vannak a képernyőn, akkor a játéktér alatt látható, az egyes emberekkel együtt mozgó monogram segítségével különböztethetjük meg őket.

Természetesen nem is lehetne a játék egy igazi Gargoyle-adventure, ha egy ilyen "egyszerű" problémával állnánk csak szembe, mint Loeg kiszabadítása. Persze a feladat nem "csak" ennyiből áll, egyéb problémákkal is szembe kell néznünk. Amíg Cuchulainn pihent, az istenek már megint a fejükre estek: kitört a teljes leszerelés, az istenek becsapták a zaciba a fegyvereket, a pénzből pedig elmentek a kocsmába. Itt természetesen összekulcsolták azon, hogy ki állja a számlát és másodperceken belül ismét beállt köztük a hadiállapot. Mivel nemcsak harcias, hanem alapvetően lusta istenek is, Cuchulainn-ra bízák, hogy időközben elkallódott pajzsait megkeresse és elszállítsa a szentélyeikbe. Az istenek haragjával nem tanácsos szembeszállni, mert előfordulhat, hogy megvárják az embert egy sötét utcán és addig csavarják az orrát, amíg olyan nem lesz, mint egy szárnyascsavár...

Ez meglehetősen sok mászkálásba és időbe kerül, mert Darach elég nagy város és feladataink teljesítéséhez majdnem az egészet be kell járunk. Ahhoz, hogy megspórolhassunk némi gyaloglást (és időt) ún. "portal"-ok állnak rendelkezésünkre, amelyeken keresztül gyorsabban eljuthatunk a város bármely távoli pontjára. Négy ilyen portal található a város négy különböző pontján, egy-egy igénybevétel 200 iridinkbe kerül (az idő pénz). A portaloikat a mellékelt térképen külön jelöltük (sötét nyíl jelzi a portálás útvonalát), és a továbbiakban is az egyes helyekre a térképen feltüntetett utca és ajtó számával fogunk hivatkozni.



Először is felfedező útra indulunk: mászkálunk egy kicsit a városban, hogy tapasztalatokat szerezzünk és kitaláljunk valamit, hogy hogyan is lássunk neki a feladatnak. Kóricálás közben elvetődünk Old Quarter negyedbe is, ahol magasszintű kulturális fórum működik: itt egy képtár (GALLERY). Ezekből a városban négy darab van és mindegyikben négy, meglehetősen szurrealista kép látható a falon. A képek között egy asztalt is találunk, ami jelzi, hogy itt valamilyen áruforgalom lehetséges. Menjünk be a Herne Hill 1-ben lévő képtárba és merüljünk el a művelzetben. A képeken különböző ábrák láthatóak, sorban: egy nyíl, egy "+Z", egy tű és egy korrel jelölt nem azonosítható dolog (mint későbbi próbálkozásaink során kiderült, egy kapa). Kóricálni fogunk egy kicsit: be kell szereznünk ezeket a tárgyakat a város kereskedőitől. Sajnos képtárunként kétszer kell fordulnunk, mert egyszerre csak 3 tárgyat hozhatunk magunkkal. Vegyünk magunkhoz némi pénzcsekkét (2-3000 iridit), majd menjünk el Graymarket merceer heath-be, ahol a nyílkesztő King Street 4. alatti boltjában veszünk 150-ért nyilat (ARROW). Ezután vegyünk - ugyanebben a negyedben - Long Lane 6-ban varrótűt (NEEDLE) a szabónál, illetve kapát (HOE) a farmernél, a Parade 4-ben. Mehetünk vissza a galériába, ahol tárgyainkat lepakoljuk a képek között lévő asztalra. A tárgyak lerakásával egy időben a hozzájuk tartozó képen egy pipa jelenik meg, ami nyugtázza a tárgy átvételét. Azt, hogy mit jelent a "+Z", némi angol tudással rendelkezők könnyen megfejthetik: "add z" kiejtve pontosan ugyanúgy hangzik, mint az ADZE (fejseje), ami a Oak Road 27-ben vehető meg a ácsnál. Miután ezt is letettük a képtárban, a sok munkánkért cserébe felvehetünk az asztalról egy ajándéktárgyat, ami egy mellűt (BROOCH). Mit lehet egy ilyen kutyuval csinálni? "Chérchez la femme!" (Keresd a nőt!) - mondja a francia, aki bár nem kelte, attól még lehet igaza. Tegyük mi is így, keressünk valamilyen feminin jellegű képződményt. Ha lehet, ne Kara vagy Keli legyen keresendő tárgya, mert ők szépen ellopják a mellűtünket, javasoljuk inkább a Pita nevű nőszemélyt. Mint a többi egyszerű élőlény, akinek kettessel kezdődik a személyi száma, ő is rendkívül kultiválta a csöcsbebecskét és ha a "SPACE" gombbal felajánljuk neki, hálából átad nekünk egy irattekercset (SCRIP). Ezen a RATS ARE VAIN felirat díszelég (A patkányok haszontalan állatok). Fantasztikus távlatokat nyit meg előttünk ez az információ, csak éppen nem tudjuk mire használni. A tekercsre nem lesz már szükségünk, tegyük le valamilyen raktárban (ezeknek a használatát ld. később).

A másik három képtárban is hasonlóan kell eljárunk: el kell hoznunk a képeken látható ábrához tartozó tárgyakat, melynek fejében újabb ékszereket kapunk. Ezeket Pita-nál becsereélhetjük újabb tekercsekre. Természetesen a képeken lévő ábrák általában csak áttételesen utalnak a beszerzendő tárgyra, így meglehetősen sok próbálkozásunkba kerül, amíg minden tárgyat idehozunk. Ezek sorban a következők:

## 2. képtár, Herne Hill 4 (Árgot)

Fullánk (STING), a Heath Road 58-ban (Graymarket) lévő gyógyszerárból  
Ólom (LEAD), a Cross Street 57-ben (Iomain) lévő alkimistától  
Burok (HEMLOCK), a Market Street 12-ben (Graymarket m.h.) lévő gyógyszerárból  
Kaptata (LAST), a Long Lane 4-ben (Árgot) lévő foltozótól

A képtárban kapunk egy fülbevalót (EARRING), amiért Pita ART IN ORDER (A művészet rendben) feliratú tekercset ad. Fogalmunk sincs, hogy mit jelenthet...

## 3. képtár, Marsh Street 34. (Ratha De)

Ruta (RUE), a Bird Street 18-ban (Ratha De) lévő fuvasztól  
Csomag (PILE), a Amber Way 65-ben (Árgot) lévő hírnöktől  
Borotva (RAZOR), a New Street 14-ben (Graymarket) lévő borbélytól  
Tőr (FOIL), a Oak Road 26-ban (Graymarket m.h.) lévő fegyverkovácsától

A képtárban kapunk egy nyakláncot (NECKLACE), amiért Pita SAYS A PEARL (Skarnak van egy gyöngye) feliratú tekercset ad. Ez nagyon fontos információ!



Nézünk meg, hogy mit sikerült eddig összehoznunk: a szentélyekben elolvasott feliratok, kiassé zagványak tűnnek (" + ST 54"), de ha figyelembe vesszük, hogy a kereszt angolul "cross" és ST = 3 az az rövidítés, már mindjárt értelem nyer a dolog - Cross Street 54. Ezenkívül Midről begyűjtöttunk egy kulcsot is (M-KEY), aminek valószínűleg szintén köze lehet a dolgokhoz. Vegyük tehát magunkhoz és nézzük meg mit van a Cross Street 54-ban (laminek): Fajdalom, ilyen ajtó találok, viszont az 52. ajtó után találok



két olyan fáklyát, ami ajtószélességben van egymástól. Ismét egy rejteajtót, amelyet - hála Midir kulcsának - kinyithatunk. Itt találhatunk egy érdekes tárgyat, a **telestone**-t. Próbáljuk meg használni valamire: nyomjuk meg a **'SPACE'**-t. Nicsak, a játéktér helyén egy másik helyet láthatunk: itt éppen egy Skar névre hallgató bájos hölgy máskéül. Nézzuk meg a utcátabán, hogy éppen merre tart. Egy idő után a **telestone** kikapcsol és ismét magunkat láthatjuk a képernyőn. Megjegyeznénk, hogy a **telestone**-t nem tudják tölték ellopni.

Émlékezzünk csak, hogy mi állt a egyik tekercsen... Skar-nak van egy gyöngye (**PEARL**). A gyöngy a **North Wall 1**-ben (**Old Quarter**) található egy asztalon, vegyük magunkhoz és jöjünk ki. Problémák vannak: a ajtó időközben bezáródott mögöttünk (**LOCKED**). Tegyük le a gyöngyöt és tapasztalhatjuk, hogy a ajtó ismét nyitva van. Tehát a gyöngyöt csak valamilyen segédeszközzel hozhatjuk ki a szobából. A **Silver Street 83**-ban (**Argot**) a Tolvajok Szövetségénél árulnak mindenféle lopási engedélyeket, nekünk is ilyenre van szükségünk a gyöngy elviteléhez. A **LICENCE** a túl drága, potom 10000 iridi. Ha nálunk van, amikor felveszük a gyöngyöt, a szoba ajtaja nem záródik be (miután kijöttünk, le is tehetjük az engedélyt). A gyöngyöt valahogy Skar-nak kéne odaadni, akit a **telestone**-on keresztül vizsgálgatva könnyen megközelíthetünk, azonban soha nem találkozzunk vele. Skar-nak ugyanis van egy apró testi hibája: láthatatlan. Egy bohó párceben ugyanis pusztá tréfából elvarázsolta önmagát is és most nem találja a varázslatot, amivel ismét láthatóvá válhat. Varázslatok kaphatóak a varázslók boltjában, a **Cinder Bank 24**-ben (**The Soke**). Itt megvásárolhatjuk a varázslatot (**SPELL**), melynek használati utasítása magától beszél: **SHOW THE LADY** (Mutasd a hölgynek!). Ezzel beugorhatunk ugyan a női toaletbe is (**LADYOUS**), de a kukkolást inkább a **PEEPSHOW** című játékbán gyakoroljuk. A varázslatot Skar-nak kell valahogy odaadni, bár a elég körülményes dolog lesz. A **telestone**-on keresztül megfigyelhetjük, hogy általában ugyanazon az útvonalon járják, ennek egyik pontján kell megvárunk. Véleményünk szerint a legcélsezerűbb a **Herne Hill** és a **Silver Street** (**Argot**) kereszteződése után várakoznunk, itt csak két irányból közelíthet felénk. A **telestone**-t sűrűn használva figyeljük, hogy merre jár és ha úgy érezzük, hogy már közelünkben van, vegyük a kezünkbe és ajánljuk fel neki a varázslatot. Ha sikerül neki odaadni, **I'M SEEN** (Látható vagyok) felirat kíséretében megjelenik testi valóságában is. Most már odaadhatjuk neki a gyöngyöt is, amiért cserébe egy pergament ad nekünk (**SCROLL**). A pergamen felirata (2\*\*25) segíteni fog nekünk Loeg kiszabadításában, tehát annyit már elértünk, hogy Skar haragja megnyhult.



Most már tehetnénk valamit Loeg kiszabadítása érdekében is... Valószínűleg nyilvánvaló mindenki előtt, hogy hol tartják fogva: a várban (**CASTLE**). A várat a **High Gate**-ről nyíló ajtón keresztül közelíthetjük meg. Itt egy információt kapunk, mely szerint a börtönőr Teth-nek hívják (**TETH** a **JAILER**). Mivel továbbmenve egy bezárt ajtót találunk, egyértelmű, hogy a bejutás kulcsa egy Teth nevű úrnál található (már amennyiben egy smasszert úrnak nevezhetünk...). Teth meglehetősen különös módon egy csatornában éldegél. Természetesen nem itt a várban, tehát mehetünk vissza a városba. Mint később kiderült, Teth amatőr műgyűjtő, vegyünk hát neki egy szobrot (**STATUE**) a **North Wall 3**-ban (**Old Quarter**). A szobor nagyon szép: egy ülő patkányt ábrázol.

A köztisztaság nem áll a legmagasabb fokon a városban, így néhány patkány is szaladgál az utcán. Jussunk eszünkbe mi volt az egyik tekercsen: "A patkányok haszontalan állatok". Ez bizonyos fókig helytálló megállapítás, de esetünkben a legfontosabb meghatározás rá: marhaság. Bár a patkány nem hajlandó elfogadni a neki felkínált, rá vonatkozó tekercset - mondván, hogy ő csak egy patkány (**I'M ONLY A RAT**)-, de azért még hasznunkra válhat. A csúnya négy lábúak a város legrejtettebb zugait is ismerik, kövessük tehát az egyiket (**The Soke**-ban a **Claw Lane-Cinder Bank-Stone Road** útvonalon szokott máskéülni egy szép példány). A patkány egy ideig vigan szalad felfelé a **Claw Lane**-en, aztán egyszer csak eltűnik. Hova a ordögbe lett? Próbáljunk csak ott átmenni a falon, ahol eltűnt. Egy újabb rejteajtót. De mi (illetve ki) van a ajtó mögött? Nem más, mint Teth, a börtönőr. Ajánljuk fel neki a patkányt ábrázoló szobrunkat, mire oromében odaadja nekünk a vár kulcsát (**D-KEY**). Ezzel a kulccsal kinyithatjuk a vár zárt ajtaját.

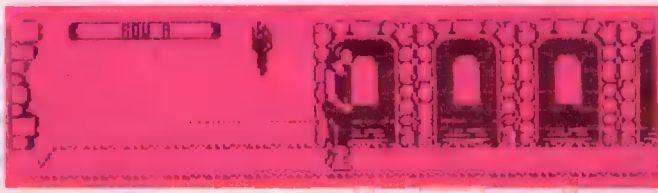
Miután átjutottunk a zárt ajtón, a változatosság kedvéért egy újabb zárt ajtóval találjuk szembe magunkat. Az ajtó mellett három asztalt találunk (valószínűleg majd le kell rakni ide valamit), amelyek felett betűk forognak. Hm, már csak egy betűzáras ajtó hiányzott a boldogsághoz. Mindössze 17576 féle lehetséges verziója van, keressuk meg inkább azokat a tárgyakat, amelyek az asztalra kellene.



A város összes élőlényét nyaggettuk már segítségért, csak harman maradtak ki a szórásból: azok, akik eddig minket nyaggettak, a három tolvaj. Most ők a maradnak ki a jóból. Természetesen ezek elég romlott emberek, jó útra kell térítenünk őket (legalább addig, amíg segítenek nekünk). Erre a legcélsezerűbb módszer a bájlat (**PHILTRE**) alkalmazása, amit a **Stone Road 73**-ban (**The Soke**) árulnak. Vigyünk mindegyik tolvajnak egy-egy adag ilyen létyőt és ajánljuk fel nekik. Kapva kapnak az üzleten, sőt mindegyikük egy-egy tekercset (**SCRIP**) ad, amelyeken a **"EVE"**, a **"DID"** és a **"POP"** feliratok láthatók. Ezeknek a kezdőbetűiből állítható be a ajtó kódja. A tekercsetek a forgó betűk alatt lévő asztalra letéve tudjuk a betűket megállítani. Megjegyeznénk, hogy a várba nem tudjuk egyszerre bevinni mindhárom tekercset, mert a bejutáshoz mindig szükség van a D-kulcsra is (ezért ha kijövünk, célsezerű valahol a közelben valamelyik raktárban elhelyeznünk). Az ajtó kinyitása után nem árt, ha bevisszük magunkkal a Skar-tól kapott pergament is, valószínű, hogy szükségünk lesz rá...

Miután bejutottunk a betűkódos ajtón, nyomaszót balsejtelmink támadnak: szemben találjuk magunkat egy ajtóval, ami NYITVA van. Mögötte valami nagyon rút cicóra számíthatunk. Ez így történik: egy 8\*8 ajtóból álló ajtó-labirintus első sorozatával találjuk szembe magunkat. Menjünk be a első és elkerülünk a vár belső térképén csillaggal jelölt helyre. Hát ez nem jó, menjünk vissza aztán próbálkozzunk a második ajtóval... eredmény ugyanaz. Egy darabig elszórakozhatnánk itt a ajtók próbálgatásával, célsezerű inkább a feliratra koncentrálni, mely szerint **"8 ROW AND BAR, FOR KEY FIND SKAR"** (8 sor és oszlop, a megoldásért keressuk Skar-t). Skar-ral már volt egy exkluzív találkánk és már kaptunk tőle egy pergament, amin a 2\*\*25 felirat szerepelt. Ez egy egyszerű hatványozás: 2 a 25. hatványán 33554432, ami azt a sorrendet jelenti, amely szerint a ajtókon (balról jobbra számolva) be kell mennünk.

Most úgy tűnhet, hogy visszakerültünk a betűkódos ajtó mögé, de a tévesszen meg senkit. Időközben a folyosó bal oldalán megjelent egy ajtó, amin Skar pergamenjének birtokában átmehetünk. Minhetetlen, de újabb ajtót találunk. Valószínűleg senki sem találja ki egyedül, hogy ez is zárva van. Megint egy betűkódot kell beállítanunk. Itt már nem segít rajtunk a Joisten se, magunknak kell kitalálnunk a kódot. Az asztalokon két könyv van, a egyikben 3 KS 4 KEY WITH LOVE (a fene tudja, mit jelenthet...), a másikon a CALL ME TO

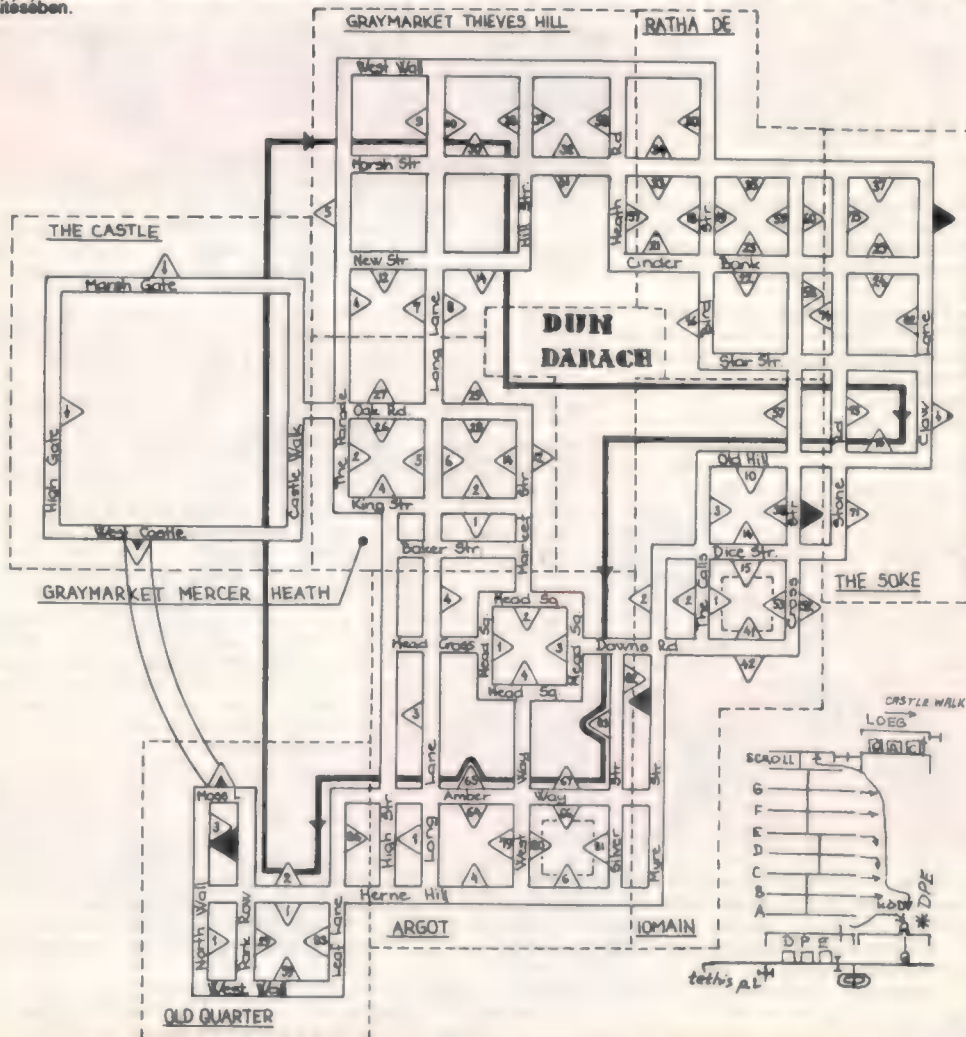




ENTER (Hívj engem ■ bejutáshoz) felirat van. Ezenkívül a harmadik asztalon egy kulcs is található (L-KEY), amelyet szépen zsebre rakunk. A könyvek ■ betűk beállítására szolgálnak, ■ harmadik betűt ■ D-kulccsal fogjuk megállítani. Lehet kezdeni ■ kód beállítását. Próbálkozzunk például annak az útnak ■ nevével, ami ■ várat összeköti ■ várossal, hátha kinyílik az ajtó...

Az ajtó mögött megtalálhatjuk barátunkat, ■ szomorkodva álldogáló Loeg-et, aki - miután odaadtuk neki ■ nevére utaló kulcsot (L-KEY) - csendben elballag. Tegyük mi is hasonlóképpen: menjünk ki ■ szobából, vegyük fel az asztalra letett D-kulcsot és szépen távozzunk ■ várból. Menjünk ki ■ várkapun és menjünk el ■ Claw Lane-en (The Soke) lévő zárt ajtóhoz, ahová ■ L-kulcs birtoklásával mehetünk be. Már teljes sötétség uralkodik ■ pályán és megjelenik a "THE END TA FROM GG" felirat ami sok mindent jelenthet... Talán azt, hogy ■ játék befejeződött és ■ Gargoyle Games gratulál nekünk, talán egy elmékedést titkosírással, ami ■ következőképpen hangozhat: "Éltem. Eldegéltem. Játékprogramokat írogattam. Megátkoztak. Többször is. Megbetegedtem. Meghaltam. Eltemettem magam. Egy könnycseppet ejtettem ■ síromra. MIÉRT NEM TUDOTT ÍGY TENNI KEITH CAMPBELL IS, MIELŐTT MEGÍRTA VOLNA EZT AZ "EGYSZERŰ" JÁTÉKOT ?!!!!"

Befejezésül szeretnénk köszönetet mondani azoknak az Olvasóinknak, akik beküldött leveleikkel segítségünkre voltak ■ leírás elkészítésében.



## Road Runner

Az örökélet bevételéhez a BASIC/6912/40960/6912/... file-térképpel rendelkező verziót választottuk ki.

Töltsük be a loader-t, RESET, majd írjuk be a következőket:

10 CLEAR 24575: LOAD ""CODE 16384: POKE 23739,111: LOAD ""CODE: POKE 40806,0: POKE 40991,0: POKE 42600,0: LOAD ""SCREEN\$: RANDOMIZE USR 34816

Futtassuk az új loader-t, majd indítsuk tovább a magnetofont.





## A Sprite kezelés elmélete

Az utóbbi időben több levelet kaptunk, melyben azt kérik tőlünk, hogy szóljunk néhány szót az **ENTERPRISE**-on megvalósítható sprite kezelés lehetőségeiről. A sprite kezelés ismertetésének két véglete van, rövid tájékoztató, és mélyreható okfejtés részletes gépi kódú mintaprogramokkal. Az utóbbi egyelőre helyszűke miatt nem fér bele a mellékletbe, de a témára természetesen még vissza fogunk térni. Így hát most következzenek az elmélet:

A sprite kezelés az akciójátékok legfontosabb alkotóeleme. Ha bármely számítógépen alakokat akarunk mozgatni a képernyőn, meg kell írunk a sprite-kezelő rutint. Ez alól csak azok a gépek a kivételek, amelyeken a manő-manipulációt a video-chip végzi, hardware úton (pl. C-64, MSX, ATARI XL, stb.).

A sprite kezelésnek is megvan a maga történelme. Az idő múlásával, ahogy a programozók fejlődtek, egyre újabb módszereket találtak ki a sprite-ok mozgására.

**Az első, legklasszikusabb példa** az, amikor az alakot kitesszük a képernyőre, majd várunk, műveleteket végzünk, s amikor már minden kész, letöröljük a képernyőt. BASIC-ben ez elég elterjedt megoldás, ennek megfelelően lassú, nehézkes és rendkívül villog.

**A második verzió** működési elve az, hogy a sprite szélein egy egységnyi széles, üres területet hagyunk. Ha ezt úgy tesszük rá az előző fázisra, hogy az elhelyezkedése csak egy egységnyivel térjen el az előzőtől, akkor a régi fázis kilógó része az üres keret darabja lesz. Ennek a módszernek az a legnagyobb hátránya, hogy a sprite-ot csak kis egységként lehet mozgatni.

**A következő lehetőség**, amikor az ábrát úgy másoljuk a képernyőre, hogy össze XOR-oljuk vele. A levétele ropant egyszerű, mindössze meg kell ismétetni az előző műveletet. Egyébként ez az első olyan módszer, amikor a sprite-kezelés két fázisban történik: *kirakás*, *letörlés*. Használata egyszerű, de az eredmény nem mindig esztétikus.

**A legszebb módszer** az ún. maszkos sprite kezelés. Két új dolgot is tartalmaz. Az egyik a *maszk*, a másik a *sprite puffer*. Egy sprite maszkolása úgy történik, hogy a sprite körvonalán belül eső résznek megfelelő biteket kioltjuk, a külső rész biteit pedig kigyűjtjük. Ezt a rajzolóprogramok FILL utasításával könnyen végre lehet hajtani.

**A sprite kezelése ■ következőképpen zajlik:**

A régi képernyőcíme vissza kell másolni a sprite puffer

tartalmát. Ez tulajdonképpen a *törlés*. Az új képernyőcíme át kell másolni a képernyő tartalmát. Ez a *tárolás*. Ki kell venni a képernyő egy byte-ját, össze kell AND-elni a maszk megfelelő byte-jával, majd 'bele' kell OR-olni a sprite adatot. Ezt azután vissza kell tenni a képernyőre. Ez a *kirakás*.

Mint látható, ez a folyamat három részből áll, emiatt lassabb is, mint az eddigiek. Azonban, ha valaki jól tud programozni, egyetlen ciklusban végrehajthatja mind a hármat, s ez jelentős időmegtakarítást jelent. Hátránya azonban az, hogy a sprite-ok nem mehetnek egymásra. Ha viszont ellentétes sorrendben tároljuk le őket, mint ahogy kitesszük, akkor ez a probléma is megoldódik. Ekkor viszont túl sok lesz az az idő, amíg sprite nincs a képen. Minél nagyobb ez az időtartam, annál jobban fog a sprite villogni. Ezt meg lehet próbálni úgy kiküszöbölni, hogy a sprite kezelést *kiszinkronizáljuk*. Ez vagy úgy történik, hogy kiadunk egy HALT utasítást, vagy úgy, hogy a sprite kezelést az interrupt rutinban helyezzük el. Mindkét esetben az elektronsugár a képernyő tetején van, ill. az **ENTERPRISE** gépen abban a raster-sorban, ahol video interrupt-ot kértünk. Ekkor, ha szerencsénk van, vagy kevés sprite-ot kellően gyors rutinnal kezelünk, a villogás elmarad. Ha nincs szerencsénk, akkor megeshet, hogy a sprite egy darabja, vagy akár az egész úgy ahogy van, eltűnik. Ez akkor következik be, amikor az elektronsugár akkor ér oda a sprite-hoz, amikor az éppen törlés stádiumban van, tehát nincs ott semmi, csak a háttér. Ez az oka egyébként az összes sprite kezelés villogásának is.

A villogás kiküszöbölésére kitaláltak egy módszert, melynek lényege a *háttérképernyő*. Ez lehet akkora, mint a rendes kép, kisebb, vagy akár nagyobb is, mindig a célnak megfelelő. Ezen kell megszerkeszteni a képet, majd a kellő nagyságú darabot, vagy akár az egészet is nagy sebességgel átmásolni az eredeti képernyőre. Ha mindez video interrupt-hoz szinkronizálva történik, akkor végre elmarad a villogás. Hátránya azért ennek is van; a program futásából sok időt vesz el a háttérkép előremásolása. Az **ENTERPRISE** gépen ezt a hibát könnyen ki lehet küszöbölni. Készíteni kell két ugyanolyan képernyőt, de kezdőcímeiket különböző memóriacímekre kell tenni. A kettőből természetesen egyszerre csak az egyik látszik. Amelyik láthatatlan, azon kell megszerkeszteni a képet, hasonlóan, mint a háttérképernyőnél. Ha a kép kész, átkapcsolhatjuk, hogy ez legyen aktív. Ezután folytathatjuk a következő fázis rajzolását a másik képen, miközben ezen látszik az előző fázis. Ezzel megtakaríthatjuk azt az időt, amit a háttérképernyő rendes képre való másolásánál elvesztenénk...



Újabb színvonalas programmal is gyarapodott az ENTERPRISE SPECTRUM átiratok köre. Egy izgalmas kalandjátékot írt át 'valaki', és nem is akárhogyan, hiszen a leginkább zokon vett attribútumhibával sem találkozunk.

A történet érdekes, **RASTAN**-t, Maradona hős királyát **Karg** a gonosz varázsló elűzte trónjáról, ezt természetesen barátunk érthető módon zokon vette, s elegendő erőt érezve ahhoz, hogy szembe szálljon legfőbb ellenségével, bőr öltözékében és öt szál pállossal felfegyverkezve elindult, hogy a varázslónak tánc- ill. illetanórákat adjon.

Mivel kedvenc mágusunk hányatott gyermekkorra óta utálja a spenótot és a tanárokat, úgy döntött, hogy szabadonereszti (jó)néhány teremtményét. A lélegzetelállító fellegrák, meredek lépcsők, tüz- és lávafolyamok között találkozunk **Karg** anyósának csontvázával (ez a legfélelmetesebb), ragadozó birkákkal (ez is azt tükrözi, hogy **Karg** sosem volt igazán jó biológiából), sőt, hogy **Raszinak** kedveskedjen, még teremtményeinek felfegyverzésén túl a királyi palotát is egy labirintusrendszerre változtatta.

Ha előre haladunk, itt-ott különféle fegyvereket, gyógyszereket és egyéb tárgyakat is találhatunk (pl. hentesbárd, buzogány, sőt még **Karg** nagymamájának egykori béli legyezője is a felvehető tárgyak listáján szerepel). Ha éppen aktuális fegyverünkkel agyoncsaptunk valakit, reménykedjünk benne, hogy a megboldogult tartogatott magánál valamiféle tárgyat, mivel annak elhalálózását követően azt a tárgyat felvehetjük. A begyűjtött cuccok a játéktér alsó részén jelennek meg, így mindig tudjuk mi van nálunk, de van egy bibi, ezek a tárgyak is rendelkeznek a fegyverekhez hasonlóan azzal a rossz tulajdonsággal, hogy csak egy bizonyos ideig vannak a birtokunkban, mivel a nagy gyűrődésben hamar elhasználódnak.

# RASTAN

## A tárgyak neve és rendeltetésük:

**SHIELD** (pajzs): némileg csökkenti a sérüléseket

**MANTLE** (köpönyeg): felére csökkenti a sérüléseket

**ARMOUR** (fegyverzet): **RASTAN** csak egy rövid ideig sérthetetlen.

**RING** (gyűrű): fokozza a fegyverek erejét

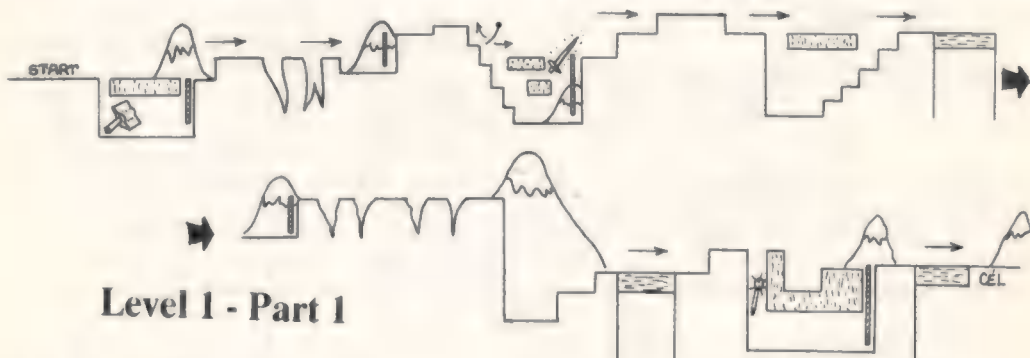
**NECKLACE** (nyaklánc): duplázza a pontok számát, és akkor is életben maradunk, ha már erről nem is álmodtunk.

**ROD** (pálca): kis tűzgolyókkal való lövöldözésre alkalmas

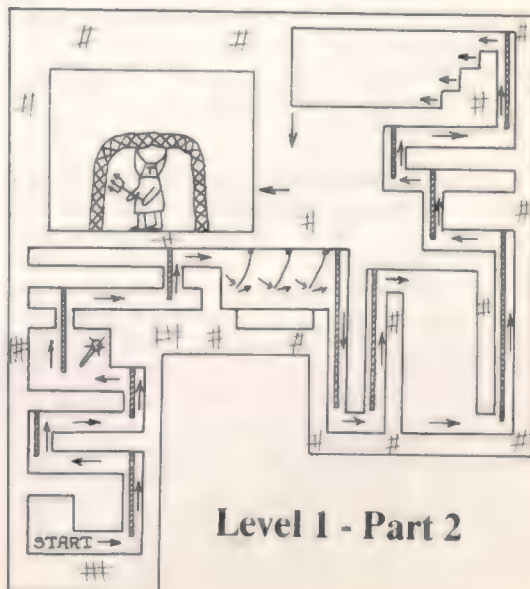
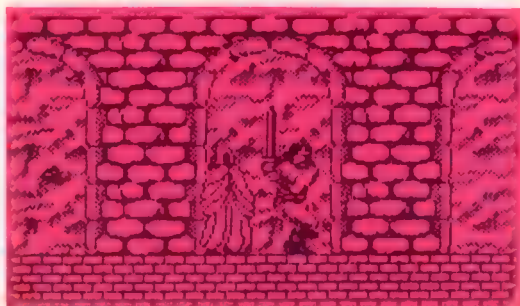
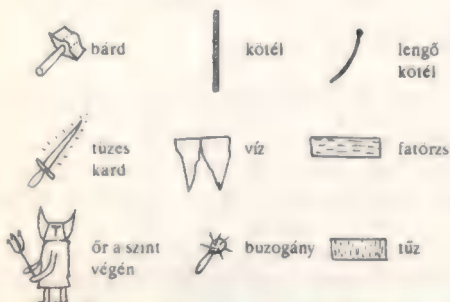
Az egyes szintek mind egy igen erős ellenfél legyőzésével tejesíthetők. A játék végén **Karg** megpróbál még 'rátenni egy lapáttal', s sárkány képében száll velünk hareba. Mi persze ettől sem rettenünk meg, legyőzzük a gonosz varázslót, akinek szelleme a fellegekbe repül, s **Maradona** népe hű szolgáknak lesz mindörökké.

A játékból a 'STOP' billentyű megnyomásával (Spectrumnál 'SPACE') léphetünk ki a menühoz a játék bármely pillanatában. A képernyő alsó részén elhelyezett sáv az energiánk mennyiségét jelzi, tőle balra a szívünk képe látható. Minél jobban dobog, annál inkább szükségünk lenne egy kis pihenésre.

A program ön maga és az átirat is magas színvonalú, a grafika szépen kivitelezett, s azok kedvéért, akik nem tudnának hozzájutni az átirathoz, meg kell említenünk azt is, hogy az általunk ismert Spectrum verzió is problémamentesen fut az ENTERPRISE SPECTRUM emulátor segítségével.

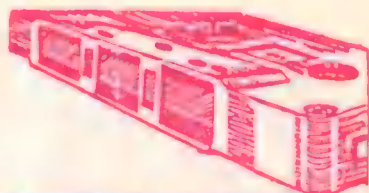
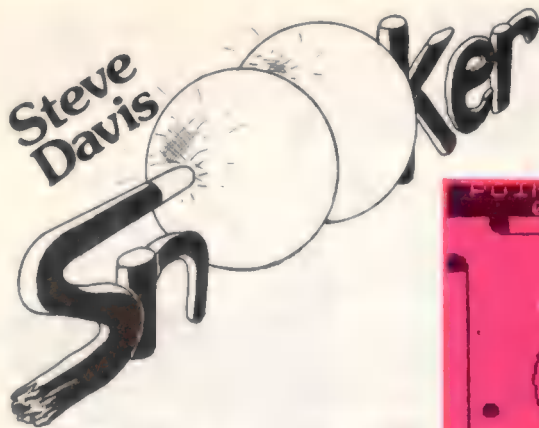


Level 1 - Part 1



Level 1 - Part 2





Ezt az **ENTERPRISE** programot a szimulációs játékok kedvelőinek tudjuk ajánlani.

A program a hatlyukú amerikai biliárd játékot szimulálja. A játék az USA-ban igen népszerű, az előkelő kasszinóktól kezdve a legkisebb italméréseken át, sőt léteznek szép számmal olyan klubok is, ahol a csontgolyók mesterei nagy téttekkel játszanak. Az is gyakran előfordul, hogy valaki a játékra CADILLAC gépkocsin érkezik, s sajnos egy cent nélkül távozik.

### A kis bevezető után induljon is a játék.

Betöltés után megjelenik a menü, melyben választhatunk:

1. Egy játékos. Ilyenkor csak egyedül játszunk.
2. Két játékos. Most ketten egymás ellen játszhatunk.
3. Céltérsezt mozgató hang bekapcsolva.
4. Céltérsezt mozgató hang kikapcsolva.
5. Hosszú játék. Így egy teljes mérkőzés lejátszására nyílik lehetőség.
6. Rövid játék. Most csak a táblán lévő golyókat kell eltüntetnünk.

A paraméterek kiválasztását követően bármely más billentyű megnyomására indul a játék.

A gép kirajzolja (viszonylag lassan) a játéktérrel. Először is el kell helyeznünk a félkörben a fehér golyót. Itt jegyezzük meg, hogy ezután mindig a fehér golyóval lökünk.

A program a kép alján jelzi, hogy milyen színű golyót kell a lyukba juttatnunk. A más-más színű golyók más-más pontszámot érnek:

Yellow (sárga)	2 pont
Green (zöld)	3 pont
Brown (barna)	4 pont
Blue (kék)	5 pont
Pink (rózsaszín)	6 pont
Black (fekete)	7 pont

Mindig csak azt a színű golyót juttassuk be a lyukak valamelyikébe, amit a gép kér, mert ha más színű is bejut, akkor az visszakerül az eredeti helyére, és növekszik a hibapontjaink száma. A hibapontjaink a következőképpen alakulhatnak:

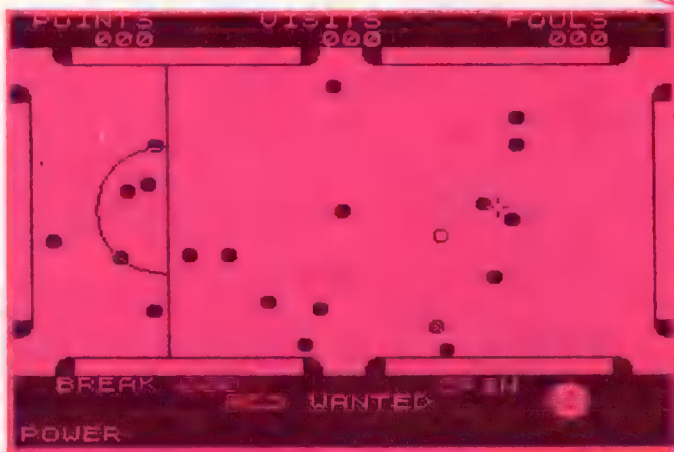
Yellow (sárga)	4 hiba
Green (zöld)	4 hiba
Brown (barna)	4 hiba
Blue (kék)	5 hiba
Pink (rózsaszín)	6 hiba
Black (fekete)	7 hiba

A fehér golyó elhelyezése után ki kell jelölnünk a lökés irányát. A céltérsezt irányíthatjuk a beépített botkormánnyal, vagy a következő billentyűkkel:

O - balra	P - jobbra
Q - felfelé	A - lefelé

**ENTER** - választás, lökés

Ha megvan az irány, nyomjuk le az **ENTER**-t. Ha ezután mégis változtatni szeretnénk, akkor a **SHIFT** és a **C** billentyűt. Most ki kell választanunk a lökés erősségét (szintén a céltérsezt vezérlő billentyűkkel), minél hosszabb a sárga csík a **POWER** felirat mellett, annál erősebbet lökünk. Ha a kiválasztás megtörtént, akkor nyomjuk le az **ENTER**-t. Ezután azt állítjuk be, hogy hol usse meg a fehér golyót a dákó (**SPIN**). Ha középre állítjuk a jelet (alfahelyzetben ott is van), akkor a golyó egyenesen halad. Ha a bal oldalát lökjük meg, akkor jobbra görbülő ívet ír le, miközben jobbra pörög, és ha utközik, a visszapatpanás is ívelt lesz. Ha ezt a bal oldali irányban tesszük, akkor természetesen ennek pont az ellenkezője fog történni. Az alsó részen



lökött golyó visszafelé pörög. A felső részen lökött golyó pedig előrefelé.

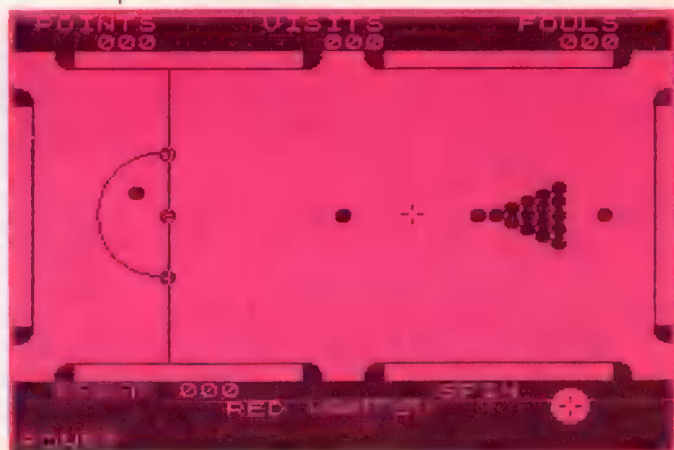
A közbelső irányokban történő lökés a fentiek kombinációját eredményezi, ezért célszerű gyakorolni, hogy kitapasztaljuk a golyó viselkedését, még mielőtt leállnánk valakivel versenyezni. Erre kiváló lehetőséget nyújt az, hogy az utolsó lökést vissza tudjuk állítani a **SHIFT** és az **R** billentyűk együttes lenyomásával.

Ha a gép a színválasztás felől érdeklődik (**COLOURS WANTED**), akkor valamelyik számbillentyű lenyomásával kell válaszolni, hogy melyik színű golyót akarjuk belökni a lyukba. A színeknek a következő billentyűk felelnek meg:

Yellow (sárga)	2
Green (zöld)	3
Brown (barna)	4
Blue (kék)	5
Pink (rózsaszín)	6
Black (fekete)	7

A játékból a menübe a **SHIFT** és az **N** billentyűk együttes lenyomásával lehet visszalépni. Természetesen ilyenkor a játék állása törlődik.

Szubjektív véleményünk, hogy a játéknak van egy nagy hibája: a fekete/fehér TV képernyőjén a fehér és a rózsaszín golyókat könnyen össze lehet tévesztetni, a zöld golyó pedig egyáltalán nem látszik, vagyis a játék ilyen készüléken játszhatatlan! A mozgás szimulációk jól sikerültek, a golyók mozgása a valóságot jól megközelíti. Szerintünk egy kis zene nem ártott volna. A programkzettához lényegretörő leírást adtak. Összegzésképpen egy játszhatóság szempontjából normális, ám grafikai és zeneileg kívánnivalókat maga után hagyó játékkal lettek gazdagabbak az **ENTERPRISE** felhasználók.







## Meglepetés!!!

```

11 SET STATUS OFF
10 DIM R(8)
11 GRAPHICS HIRES 16
12 SET PALETTE 0,0,0,0,0
13 SET LINE MODE 0
14 PLOT 0,0,40,0;84,48,60,68,0,0
15 PLOT 50,40,PAINT
16 PLOT 0,128,48,156;38,176,0,160,0,128
17 PLOT 20,160,PAINT
18 PLOT 0,260,32,280;32,288,0,288,0,260
19 PLOT 20,280,PAINT
20 PLOT 0,388,28,372,40,400,0,416,0,388
21 PLOT 30,400,PAINT
22 PLOT 0,532,60,476,76,496;32,540,0,540,0,532
23 PLOT 40,520,PAINT
24 PLOT 170,0,204,0,212,36;184,36;170,0
25 PLOT 200,30,PAINT
26 PLOT 284,0,308,0,308,36;284,36,284,0
27 PLOT 300,20,PAINT
28 PLOT 392,0,416,0,408,36;360,36,392,0
29 PLOT 400,20,PAINT
30 PLOT 566,0,584,0,584,12,536,60;512,40,556,0
31 PLOT 560,20,PAINT
32 PLOT 580,128,580,160;560,172;548,148,580,128
33 PLOT 560,160,PAINT
34 PLOT 584,264;584,288,556,288,556,284,584,264
35 PLOT 560,280,PAINT
36 PLOT 584,392;584,420;540,400;552,376,584,392
37 PLOT 560,400,PAINT
38 PLOT 584,540;540,540,508,500;524,480,584,540
39 PLOT 540,520,PAINT
40 PLOT 400,540;376,540,388,508,392,504,400,540
41 PLOT 380,520,PAINT
42 PLOT 300,540;276,540;276,512;300,512;300,540
43 PLOT 280,520,PAINT
44 PLOT 196,540;172,540;180,504,204,512;196,540
45 PLOT 185,520,PAINT
46 SET INK 2
47 PLOT 80,88,84,48;124,92;104,108;60,88
48 PLOT 96,80,PAINT
49 PLOT 48,156;80,172;88,196;36,176;48,156
50 PLOT 80,180,PAINT
51 PLOT 32,260,60,260,60,288,32,288;32,260
52 PLOT 40,280,PAINT
53 PLOT 80,356;72,384;40,400,28,372;60,356
54 PLOT 40,380,PAINT
55 PLOT 80,476;104,432;120,452;76,496,60,476
56 PLOT 80,480,PAINT
57 PLOT 204,512;180,504;192,468,216,476;204,512
58 PLOT 200,480,PAINT
59 PLOT 276,476;300,476;300,512;276,512;276,476
60 PLOT 280,480,PAINT
61 PLOT 368,508;360,476;384,468,392,504;368,508
62 PLOT 380,480,PAINT
63 PLOT 504,500;480,456;480,436,520,480;504,500
64 PLOT 500,480,PAINT
65 PLOT 540,400;504,380;516,356,552,376,540,400
66 PLOT 520,380,PAINT
67 PLOT 556,288;528,268,528,264,556,264,556,288
68 PLOT 540,280,PAINT
69 PLOT 528,188;516,188;548,148,560,172;528,188
70 PLOT 540,160,PAINT
71 PLOT 488,108;472,92;516,44,536,80,488,108
72 PLOT 500,80,PAINT
73 PLOT 408,36;400,68;372,68;380,36,408,36
74 PLOT 380,60,PAINT
75 PLOT 308,36;308,58;284,68;284,36,308,36
76 PLOT 288,80,PAINT
77 PLOT 212,36;220,98;192,68;184,36;212,36
78 PLOT 200,40,PAINT
79 SET INK 3
80 PLOT 124,92;160,128,140,148;104,108;124,92
81 PLOT 140,120,PAINT
82 PLOT 80,172;108,184,96,212,68,196,80,172
83 PLOT 84,166,PAINT
84 PLOT 80,260,80,260,80,288,60,288;60,260
85 PLOT 72,280,PAINT
86 PLOT 80,356;88,340,100,364;72,384,60,356
87 PLOT 80,380,PAINT
88 PLOT 136,400;148,420;120,452;104,432;136,400
89 PLOT 120,425,PAINT
90 PLOT 216,476;192,488;200,440,224,444,216,476
91 PLOT 200,460,PAINT
92 PLOT 500,448;300,476;276,476,276,448,300,448
93 PLOT 280,460,PAINT
94 PLOT 356,448;380,444;384,488,360,476,356,448
95 PLOT 360,460,PAINT
96 PLOT 444,400;480,436;460,458,428,420,444,400
97 PLOT 440,420,PAINT
98 PLOT 488,344;516,356;504,380;480,364;488,344
99 PLOT 500,360,PAINT
100 PLOT 500,284;528,264;528,288,500,288,500,284
101 PLOT 520,280,PAINT
102 PLOT 516,188;528,188,496,208,484,184,516,188
103 PLOT 500,180,PAINT
104 PLOT 472,92;488,108,452,144;432,128,472,92
105 PLOT 480,120,PAINT
106 PLOT 400,68;384,104,380,100;372,68,400,68
107 PLOT 380,60,PAINT
108 PLOT 308,68;308,100,284,100,284,68;308,68
109 PLOT 288,80,PAINT
110 PLOT 220,68;226,100,204,108;192,68;200,68
111 PLOT 204,80,PAINT
112 SET INK 4
113 PLOT 168,120;132,156;180,164;168,120
114 PLOT 160,140,PAINT
115 PLOT 112,160;92,224;136,220;112,180
116 PLOT 110,210,PAINT
117 PLOT 80,248,124,276,80,300;80,248

```

```

118 PLOT 100,270,PAINT
119 PLOT 84,328;112,376;136,332;84,328
120 PLOT 120,340,PAINT
121 PLOT 128,392;160,428;172,380,128,392
122 PLOT 160,400,PAINT
123 PLOT 228,408;242,458;192,428;228,408
124 PLOT 200,480,PAINT
125 PLOT 284,448;268,412;308,448,264,448
126 PLOT 280,440,PAINT
127 PLOT 340,452;356,408;392,436;340,452
128 PLOT 380,420,PAINT
129 PLOT 420,428;412,384;452,392;420,428
130 PLOT 420,400,PAINT
131 PLOT 472,372;452,336;492,332,472,372
132 PLOT 460,340,PAINT
133 PLOT 504,300,464,276;504,252;504,300
134 PLOT 482,280,PAINT
135 PLOT 500,280;500,284
136 PLOT 504,220;456,216,476,176;504,220
137 PLOT 480,200,PAINT
138 PLOT 424,124;416,164,460,148;424,124
139 PLOT 430,144,PAINT
140 PLOT 348,96;360,136;396,108;348,96
141 PLOT 360,120,PAINT
142 PLOT 272,100,296,140,316,100;272,100
143 PLOT 300,120,PAINT
144 PLOT 240,96,192,112,224,140,240,96
145 PLOT 220,120,PAINT
146 SET INK 5
147 PLOT 340,280,ELLIPSE 110,110
148 PLOT 332,280,ELLIPSE 110,110
149 SET INK 6
150 PLOT 324,280,ELLIPSE 110,110
151 SET INK 5
152 PLOT 316,280,ELLIPSE 110,110
153 PLOT 308,280,ELLIPSE 110,110
154 SET INK 0
155 PLOT 300,280,ELLIPSE 110,110
156 SET INK 5
157 PLOT 292,280,ELLIPSE 110,110
158 PLOT 284,280,ELLIPSE 110,110
159 SET INK 0
160 PLOT 276,280,ELLIPSE 110,110
161 SET INK 5
162 SET INK 0
163 FOR A = 110 TO 0 STEP -1
164 PLOT 268,280,ELLIPSE A,A
165 NEXT
166 SET INK 5
167 PLOT 268,280,ELLIPSE
168 PLOT 268,280,ELLIPSE 110,110
169 PLOT 190,280,PAINT
170 SET INK 0
171 FOR A = 250 TO 310
172 PLOT 268,A,460,A
173 NEXT A
174 FOR T = 0 TO 10
175 SET INK 0
176 PLOT 64,630
177 SET INK 6
178 PRINT #101, CENTRUM
179 SET PALETTE 0,137,0,0,0,219
180 FOR A = 0 TO 40
181 NEXT
182 SET PALETTE 0,137,137,0,0,219
183 FOR A = 0 TO 40
184 NEXT
185 SET PALETTE 0,0,137,137,0,219
186 FOR A = 0 TO 40
187 NEXT
188 SET PALETTE 0,0,137,137,0,219
189 FOR A = 0 TO 40
190 NEXT
191 NEXT
192 FOR A = 1 TO 8 STEP 1
193 SET PALETTE 0,0,0,0,0,RGB(A,2,0),219
194 FOR O = 0 TO 20
195 NEXT O
196 NEXT A
197 SET PALETTE 0,137,137,137,137,219
198 WAIT DELAY 10
199 LET R(1) = 60:LET R(2) = 103:LET R(3) = 125:
LET R(4) = 93:LET R(5) = 205:LET R(6) = 57
200 DO
201 LET X = R(6)
202 FOR A = 5 TO 1 STEP -1
203 LET R(A+1) = R(A)
204 NEXT A
205 LET R(1) = X
206 SET PALETTE 0,R(1),R(2),R(3),R(4),137,219
207 LOOP

```



# Micro PROLOG

A gyűjtőszendevély sok Spectrumos birtokába juttatott olyan programokat is, amelyek használatáról fogaimuk sincs, sohasem használták, azt ■■■■ tudják, mire való.

Erre ■ sorsra rendeltetett - méltatlanul - ■ **Micro-PROLOG** interpreter is. A helyzet érthető, hiszen maga ■ **PROLOG** nyelv ■■■■ túlságosan kozmikum (még ■ számítástechnikával hivatásszerűen foglalkozók körében sem), mit várhatunk akkor ennek egy nyelvjárásától?

A jövő a **PROLOG**-é, illetve örökösei. A japán kutatások az ötödik generációs számítógépek kialakításában alapvető szerepet tulajdonítanak ■ nyelvnek. Az információs társadalom kialakításában nagy jelentősége van ■ mesterséges intelligenciát felhasználó, számítástechnikán alapuló rendszereknek.

A számítástechnika fejlődése fokról-fokra lehetővé teszi ■■ eddig emberek által végzett szellemi munka automatizálását is. Intelligenciát igényel egy feladat megoldása, ha nem adható meg rá 'jó' algoritmus. Pontos meghatározása ■■ mesterséges intelligenciának éppen ■ 'jóság' definíciatlanságából fakadóan nincsen, inkább csak korlátozásokkal élnek ■■ területtel foglalkozók. Általában ■ tárgyakok és módszerek felsorolásával próbálják behatárolni ■■ mesterséges intelligencia területét. A tárgyakok ■■ következők: tételbizonyítás, analízis, tudástechnológia és még sok más. A legfontosabb módszerek: problémareprezentáció, keresési eljárások, következtetési technikák, tudásreprezentáció és tudáshasznosítás.

Az ■■ tárgykörbe vágó kutatások az 1950-es években kezdődtek meg, egyrészt ■ matematikai tételbizonyítások, másrészt ■ logikai igényítő játékok (sakk, bridge, go, stb.) területén. Az első témakörben születtek ■■ mélyebb, ■■ másodikban ■■ látványosabb eredmények. Az 1960-as években csalódást, és ■■ kutatások visszafogását eredményezte az ■■ felismerés, hogy ■■ leginkább várt (és beigért) eredmény, a természetes nyelvek fordításának megoldása jóval nehezebb, mint ahogy számították. A 70-es években ■■ számítástechnika lehetőségeinek bővülésével egyre újabb területek kerültek ■■ mesterséges intelligencia kutatások vonzáskörébe, mások viszont (pl. kézzel írt szövegek olvasása) kikerültek onnan, mivel sikerült 'jó' algoritmust találni megoldásukra. A 80-as években ■■ szakértői rendszerek, az emberi szakértelmet, tudást tároló, gyűjtő, következtetéseket levonó, tanácsadó rendszerek látványos fejlődésének, ■■ mikroszámítógépeken való futtathatóság folytán jelentős piaci sikereinek is tanúi lehettünk.

A mesterséges intelligencia kutatások egyik - Európában és Japánban - legelterjedtebb számítógépes nyelve ■■ **PROLOG**. A másik alapnyelv, ■■ **LISP**, főként az USA-ban foglal el jelentős pozíciókat. (A **SPECTRUM**-ra természetesen **LISP** Interpreter is készült, ezzel később fogunk foglalkozni.)

Míg a **LISP** egy hagyományosabb, bár a szimbolikus számítások érdekében kidolgozott funkcionális nyelv, addig ■■ **PROLOG** már onmagában is egy szakértői rendszerváznak tekinthető, amelybe egy tudásbázis kezelő és egy következtető rendszer van beépítve. Számos egyéb mesterséges intelligencia nyelv létezik még, ezek általában ■■ két alapnyelvre, vagy azok kombinációjára épülnek. Természetesen hagyományos nyelvek is használhatóak - szerepel ■■ irodalomban **BASIC** és **FORTRAN** nyelvek írt szakértői rendszer is, nem beszélve az ■■ területen is gyakori **FORTRAN** alkalmazásokról - csak éppen egészen más hatásokkal. Ez nyilvánvaló, hiszen ezekben ■■ nyelvekben azokat az eszközöket is el kell készíteni, amit ■■ **LISP** illetve ■■ **PROLOG** beépítve nyújt ■■ felhasználóknak.

Az első **PROLOG** implementációt 1972-ben készítették Marseille-ben, Alain Colmerauer vezetésével, elméleti alapjait a logikai programozás bázisán Robert Kowalski dolgozta ki Edinburgh-ben. A nyelv sikerének és elterjedésének egyik jelentős tényezője volt ■■ 1970-es évek közepén DEC System-10 számítógépre

kialakított hatékony **PROLOG** implementációja, amelyhez ■■ későbbi változatok is igyekeztek alkalmazkodni. Magyarországon már 1975-ben készült **PROLOG** Interpreter (Szeredi-Futó), ezt ■■ CDL-ben írt rendszert több nagygépen is implementálták. Ez szolgált alapjául ■■ későbbiekben kifejlesztett **MPROLOG** (Moduláris **PROLOG**) nyelvnek, amely az IBM-PC személyi számítógépeken is elérhető.

Napjainkban számos **PROLOG** változat van ■■ piacon nagy- és mikrogépes területen egyaránt. (IBM-PC-re pl.: Arity **PROLOG**, Turbo **PROLOG**, Alvey **PROLOG**, IBM **PROLOG**, micro **PROLOG**, Professional **PROLOG**, P2 **PROLOG** stb.)

A micro-**PROLOG** 1981-82-ben jelent meg, mint a világ első mikroszámítógépes **PROLOG** változata. Kifejlesztője ■■ K.L. Clark és F.G. McCabe (ismert mesterséges intelligencia kutatók) alapította **LPA** (Logic Programming Associates) társaság. Először Z-80 alapú CP/M gépekre, majd Spectrum, BBC Acorn, QL, Commodore-64 stb. gépekre implementálták. A későbbiekben MS-DOS ill. Unix operációs rendszerek alatt futtatható továbbfejlesztett változatok készültek professzionális személyi számítógépekre. A micro-**PROLOG** leszármazottja ■■ **Professional PROLOG**, amely interpretert, compiler-t és programfejlesztő rendszert is tartalmaz.

E feltétlenül elengedhetetlen bevezetőt követően ismertetjük a **SPECTRUM** micro-**PROLOG** interpreter használatának alapjait, igyekezőn bemutatni ■■ nyelv szellemét is. Célunk az érdeklődés felkeltése és elindulási fogódzók adása. Felhívjuk a figyelmet arra, hogy a micro-**PROLOG** **SPECTRUM** és IBM változatáról 1989 első negyedévében könyv is napvilágot lát az I.SIATSz. kiadásában.

## SPECTRUM Micro-PROLOG (LPA)

### Betöltés és ismerkedés

A micro-**PROLOG** kazettájának mindkét oldalán először ■■ **PROLOG** interpreter (értelmező program) foglal helyet (az S80 kollekción csak ez szerepel). Ezt követően ■■ micro-**PROLOG** nyelvű bővítések, utility-k következnek. Az 'A' oldalon ■■ **SIMPLE** és ■■ hozzá kapcsolódó modulok, ■■ B oldalon a **MICRO** és kiegészítései vannak.

Válasszuk az A oldalt és töltsük be ■■ micro-**PROLOG**-ot ■■  
LOAD "PROLOG", vagy ■■  
LOAD"" utasítással.



A **BASIC** nyelvű program először egy képet tölt be a képernyőre, majd magát ■■ zommal gépi kódban megírt micro-**PROLOG** interpretert. A sikeres beolvasás után ■■ következő üzenet jelenik meg:

SPECTRUM micro-**PROLOG** T1.0  
(c) 1983 LPA Ltd.  
24153 Bytes Free  
& L



A **■** jel mutatja, hogy **■** rendszer fő moduljában vagyunk. A (pont) pedig jelzi, hogy a program valaminek **■** beírását várja. A villogó 'L' betű (kisbetű) kurzor a szokásos módon átváltható 'C' (nagybetű), vagy 'E' (kiterjesztett) módba. (K (kulcsszó) módba nem kapcsolható, mert a **micro-PROLOG** a Spectrum BASIC utasításait nem értelmezi.)

A program betöltése után a keret zöld, a képernyő pedig sárga. Egyszínű monitorral vagy fekete-fehér TV-vel rendelkezők számára ajánlható ezek fehérre állítása:

& BORDER 7 ENTER  
&CLS 7 ENTER

A fenti utasításokat karakterenként, nagybetűkkel kell beírni. **■** **micro-PROLOG** megkülönbözteti **■** kis- és nagybetűket - és az ENTER gomb megnyomásával kell elküldeni (a későbbiekben **■** ENTER gomb használatát nem jelezzük). A beírt és még el nem küldött szövegek **■** Spectrumon szokásos módon javíthatók. A **BORDER** és **CLS** parancsokkal **■** keret és **■** háttér színét állíthatjuk.

A **micro-PROLOG** utasításkészlete nem rögzített, **■** programok írása tulajdonképpen ezek bővítését jelenti. Természetesen adott egy alapkészlet, ezeket rendszerutasításoknak, pontosabban rendszer relációknak nevezzük. Mint **■** későbbiekben látni fogjuk, a **PROLOG** (PROGRAMMING in LOGic, programozás logikában) **■** logika kifejezéseit használja, s alapvető fogalma az objektumok közötti kapcsolat, **■** reláció. Azon objektumokat, amelyeket kapcsolatba hozunk egymással, **■** reláció argumentumainak nevezzük. Egy reláció definiálásakor más felhasználói vagy rendszer-relációkra támaszkodunk. Távolról hasonló ez **■** **FORTH** szavak megadására, még távolabbról a névvel hívható szubrutinok írására.

### Kísérletezzünk egy kicsit:

& PP SZIA  
SZIA  
&P ember  
ember& LIST DICT  
((DICT & () ) ember SZIA "ERROR?")  
& SPACE x  
& LIST DICT  
((DICT & () ) "ERROR?")

A PP (Pretty Print, csinos kiírás) kiírja az utána következő kifejezést és sort emel, míg a P ugyanezt soremelés nélkül végzi. A LIST kilistáz egy relációt; most **■** program szótárát íratjuk ki vele. Látható, hogy minden eddig beírt új szó megjelenik itt. A SPACE x segítségével **■** már nem használt terület szabadítható fel. Az utána következő LIST DICT mutatja az eredményt: saját szavaink töröltek.

Ha valami miatt szeretnénk tiszta lappal újakezdeni mindent, ezt **■** NEW paranccsal érhetjük el. Ne felejtsük el beírni **■** (pont)-ot!

&.NEW

SPECTRUM micro-PROLOG T1.0  
(c) 1983 LPA Ltd.  
24153 Bytes Free

&.L

A betöltés utáni alapállapothoz jutottunk, eltekintve **■** megváltoztatott színektől.

### A micro-PROLOG alapvető szabályai

Minden eddig begépelte szövegünk két részből állt: egy reláció névből és egy argumentumból. Ha ettől eltérünk, hibauzenetet kapunk.

& PP SZIA haver krapek  
SZIA  
&.ERROR 2  
&.

Azt persze megtehetjük, hogy egy sorba több szabályos, két kifejezésből álló utasítást írunk:

& P SZIA P Haver PP krapek  
SZIA& Haver&krapek  
&.

Van arra is lehetőség, hogy több argumentumos relációt használjunk; ehhez **■** rendszer ? relációját kell igénybe vennünk:

& ?((PP Szervusz kérlek alássan))(PP))  
Szervusz kérlek alássan

&.

A ? reláció körülbelül azt jelenti, hogy 'hajtsd végre!', vagy 'No, mi lesz ebből?'. Ennek argumentuma egy listából álló lista. Egy lista **■** **micro-PROLOG**-ban zárójelek közé írt kifejezések helyekkel (vagy más módon) elválasztott sorozata. A ? hatására az argumentumában (egy listában) szereplő összes részlista - itt: (PP SZIA haver) és (PP) - egyenként végrehajtódik mint reláció. A lista formában már tetszőleges számú argumentuma is lehet **■** relációknak. (A fenti példában **■** elsőnek 2, **■** másodiknak 0 argumentuma volt.) Természetesen az egyargumentumú relációt is használhatjuk lista formában:

& ?((PP SZIA))  
SZIA  
&.

### A SIMPLE bővítés

A **micro-PROLOG** bővíthető, akár teljesen újírható nyelv. A programkzettán számos előre elkészített bővítése, 'segéd-programja' található meg. Az A oldalon, közvetlenül **■** PROLOG után van felvéve **■** SIMPLE nevű, **micro-PROLOG**-ban megírt program. Ez egy nagyon kényelmesen használható, az angol nyelvre hasonlító szintaxisal látja el a **micro-PROLOG**-ot. (MÁS PROLOG rendszerekhez leginkább ez a bővítés áll közel.) Hasonló nyelvkiegészítést elvileg minden felhasználó készíthet, persze ehhez kemény munka kell.

### Töltsük be a SIMPLE-t!

&.LOAD SIMPLE

vagy

& LOAD "SIMPLE"

A 26 fejnélküli blokkból álló program mintegy 1 perc alatt olvassódik be. Minden blokk sikeres beolvasását külön jelzi:

SIMPLE 01 BLOCK OK

Olvadási hiba esetén a következő üzenet jelenik meg:

SIMPLE 07 READ ERROR

Feltételezve, hogy a 07. blokk beolvasása volt hibás. Ilyenkor térjünk vissza **■** szalagot és újra próbálkozzunk **■** blokk beolvasásával. Ha sehogyan nem sikerül, érdemes megszakítani BREAK-kei a beolvasást (SYMBOL SHIFT + SPACE), majd NEW. után kezdhetjük előről.

### Programozás SIMPLE-ben

A PROLOG-ban - kissé leegyszerűsítve a dolgokat - közölni kell **■** rendszerrel bizonyos tényeket, majd kérdéseket tehetünk fel. A



rendszer ■ rendelkezésére bocsájtott ismeretanyag alapján fog válaszolni nekünk.

A SIMPLE betöltése után ■ a következőképpen történhet:

```
&.add(Endre apja Ferenc)
&.add(Ede apja Endre)
&.add(Zsolt apja Endre)
&.add(Kelemen apja Zsolt)
&.
```

Az 'add' egyargumentumos (mégpedig lista ■ argumentum) reláció segítségével új ismereteket adhatunk ■ rendszernek. A megadott információkat áttekinthetjük:

```
&.list all
Endre apja Ferenc
Ede apja Endre
Zsolt apja Endre
Kelemen apja Zsolt
&.
```

Most már kérdezhetünk:

```
&.is(Ede apja Endre)
YES
&.is(Endre apja Zsolt)
NO
&.is(Kelemen apja Zsolt ■ Zsolt apja Endre)
YES
&.
```

Az 'is' fennáll reláció eldönti egy állításról, hogy - legalábbis ■ rendszer ismeretei szerint - igaz-e. Ha igaznak találja: 'YES'-t ír ki, ha nem: 'NO'-t. Más is lehet kérdezni: a 'which' (melyik) relációval bizonyos feltételeknek eleget tévő objektumokat választ-hatunk ki:

```
&.which(x:Ferenc apja x)
NO (more) ■■■■■■
&.which(x:x apja Endre)
Endre Ferenc
Ede Endre
Zsolt Endre
Kelemen Zsolt
NO (more) ■■■■■■
```

A NO (more) answers (NINCS (több) válasz) jelzi, hogy a rendszer ismeretei a kérdéssel kapcsolatosan kimerültek.

Érdekes lenne megtudni, hogy ki kinek ■ apai nagyapja. Nyilván akkor x apai nagyapja y. Ha ■■n olyan z, aki ■ ■ apja és egy-üttal y pedig ■ apja, adjuk ezt ■■ összefüggést ■ rendszer tudtá-ral

```
&.add(x apai_nagyapja y if ■ ■ apja ■ and z apja y)
&.
```

Most már kérdezhetünk:

```
&.is(Ede apai_nagyapja Ferenc)
YES
&.is(Endre apai_nagyapja Zsolt)
NO
&.which(x y:x apja Ferenc and y apai_nagyapja x)
Endre Zsolt
NO (more) answers
```

Az eddigiekhez hasonlóan tényekkel és összefüggésekkel bőví-ve ■ rendszer tudásbázisát, egyre bonyolultabb kérdések megol-dására tehetnénk alkalmassá. A PROLOG programozás éppen ebből áll: megfelelő tényállításokat és összefüggéseket kell ■ rendszernek megadni, hogy ■ kérdéseinkre válaszolni tudjon.

## Szerkesztés SIMPLE-ben

A bevitt összefüggéseket a 'list' reláció segítségével írhatjuk ki.

```
&.list apai_nagyapja
X apai_nagyapja Y if
  X apja Z and
  Z apja Y
&.
```

A 'list all' ■ összes reláció információját felsorolja. A 'list' reláci-ónév csak egy relációhoz tartozó ismereteket listázza. A 'delete' segítségével törölhetjük egy reláció valamelyik összefüggését:

```
&.delete apja 2
&.
```

Ezzel az utasítással kitöröltük ■ 'apja' reláció 2. összefüggését, azt, hogy Ede apja Endre. Listázásánál már nem fog szerepelni: 'elfelejtette'.

Megszüntethetünk egy relációt teljes egészében is:

```
&.kill apai_nagyapja
&.
```

Javítani ■ relációk egyes kijelentéseit ■ következőképpen lehet: edit relációnév sorszám, pl.:

```
&.edit apja 1
1 (Endre apja Ferenc)
```

A képernyőn megjelent szöveget ■ Spectrumon szokásos módon javíthatjuk, majd ENTER-rel elküldve az átalakított kijelentés ke-rül ■ eredeti helyére.

## Szalagra mentés

A rendszerbe bevitt állítások, összefüggések későbbi felhasznál-ás ill. a továbbfejlesztés érdekében szalagra menthetők:

```
&.save "papa"
```

vagy

```
&.save papa
```

Ezeknek ■■ utasításoknak ■ hatására a:

```
start tape for recording
HIT ENTER when ready
```

üzenet jelenik meg. (Indítsuk el ■ magnót ■ felvételhez és nyom-juk meg ■■ ENTER gombot, ha minden rendben van.) A felhaszná-lói programok ugyanolyan blokk-szerkezetben íródnak ki, mint pl. ■ SIMPLE vagy a többi, ■ rendszerrel adott bővítés.

A 'save' reláció ■ SIMPLE része, csak annak betöltése után hasz-nálható. A 'save' utasítás ■■ alaprendszer része, mindig rendelkezésre áll, de nem küld figyelmeztetést és néhány másodperc múlva automatikusan elkezd ■■ kimentést:

```
&.save "papa"
```

vagy

```
&.save papa
papa 01
papa 02
```

Egy korábban kimentett programot, vagy ■■ alaprendszer már



megismert 'load'-jával, vagy ■ SIMPLE hasonló 'load' utasításával lehet visszatölteni.

§.load papa

## Záró megjegyzések

A **micro-PROLOG**-ból ■ **BASIC**-be jutni csak ■ számítógép újraindításával (RESET) lehet.

A billentyűk karakterkiosztása a szokásos, eltekintve ■ 'K' (kulcszó) üzemmód illetve bizonyos SYMBOL SHIFT ill. 'E' (kiterjesztett) módban található BASIC kulcsszavak hiányától.

Megszakítást (BREAK) ■ Symbol SHIFT + SPACE, listázás fel-függesztését ■ Symbol SHIFT + 'A', ■ képernyőre kerülő Output egyidejű kinyomtatásának megkezdését és abbahagyását ■ Symbol SHIFT + 'P' kombinációkkal lehet elérni.

A 'G' (grafikus) üzemmód is a szokásos, segítségével a felhasználói karaktereket pl. ékezetes betűket is alkalmazhatunk programjainkban.

### A **micro-PROLOG** kazettáján lévő programok:

A.oldal	B.oldal
PROLOG(P,C,C)	PROLOG (P,C,C)
SIMPLE (26)	MICRO (18)
SIMTRACE (13)	EXPTRAN (6)
EXPTRAN (6)	TOLD (3)
TOLD (3)	MICSHOW (2)
SIMSHOW (2)	EDITOR (21)
PROGRAM (8)	MODULES (5)
DEFTRAP (6)	TRACE (9)
SPYTRACE (4)	SPYTRACE (4)
ERRTRAP (6)	ERRTRAP (6)

A programok mellett zárójelben a file-ok típusa (P=program, C=kód) illetve a fejrőlki blokkok száma áll.

## Néhány megjegyzés

A **PROLOG** egyik legfontosabb tulajdonsága, hogy leíró jellegű (deklaratív). Nem azt kell egy problémáról tudnunk, hogy miképpen oldható meg, hanem azt, hogy mivel jellemezhetjük a megoldást. Ugyanúgy, ahogy egy vendéglőben sem az elfogyasztani kívánt ételek receptjét közöljük rendeléskor, hanem ■ nevüket, s az elkészítést rábizzuk ■ szakácsra, ■ **PROLOG**-ban is

■ 'MIT' az érdekes, ■ 'HOGYAN' nem. (Ha mégis bele akarunk szólni ■ vendéglőben és ■ **PROLOG**-ban is lehetőségünk van rá.)

Hogyan kerülhető ki ■ feladat algoritmusának megadása? Úgy, hogy ■ **PROLOG** rendelkezik két alapvető mechanizmussal, a mintaillesztéssel és az automatikus visszalépéses kereséssel.

Ezek azok az általános eljárások, amelyek segítségével elvileg minden feladat megoldható, legfeljebb néhány esetben ■ megoldás kissé sokáig tart.

Jól látható, hogy csak arra kaphatunk választ, ami az adatbázisban is van írva, vagy abból (az adatbázis részét alkotó szabályok segítségével) kikövetkeztethető. Ezt az alapállást szokták zárt világhipotézisnek nevezni.

Több megoldás esetén a rendszer mindegyiket megkeresi, ez ■ jelentése annak, amikor azt mondják, hogy ■ **PROLOG** nem determinisztikus.

A **PROLOG** alapvető összetett objektuma a lista, ennek segítségével különösen jól kezelhetők ■ fa-szerkezetű adat-struktúrák.

Fontos még megemlíteni ■ rekurzitást. Ennek segítségével bizonyos problémákat rendkívül tömören lehet megfogalmazni.

A most felsorolt előnyös tulajdonságai mellett a **PROLOG** lényegében egyetlen nagy hátránnyal rendelkezik: ■ feladat lineáris növekedésével exponenciálisan nőhet a megoldáshoz szükséges tárterület és futási idő. Ezt ■ tulajdonságot szokás kombinatorikai robbanásnak nevezni.

A Spectrum memória korlátai persze nem engedik meg monstre programok írását, viszont ■ **micro-PROLOG** tömörsége lehetővé teszi egyszerű, de jól működő, szakértői rendszerek kialakítását, pl. elkészíthető és ismeretekkel feltöltve használható orvosi diagnosztikai rendszer is. Ennek minősége lényegében csak annak ■ orvosnak ■ tapasztalataitól, ismereteitől függ, aki 'betanítja' a rendszert.

Kézenfekvő felhasználási lehetősége a **micro-PROLOG**-nak az adatbázis-kezelés. Minden további nélkül alkalmazható pl. szótár-programként. Egyszerűen fel kell tölteni a rendszert ■ **SIMPLE**-ben (one jelentése egy) (two jelentése kettő) stb. tényállításokkal, s utána könnyen visszakereshetjük ■ szavakat mindkét irányban: which(x: x jelentése egy) illetve: which (x: two jelentése x). (Ha a különböző kiadványokban megjelent szótár programokat megvizsgáljuk, egyik sem kínál ilyen kényelmes lehetőségeket, főként pedig továbbfejlesztési lehetőségeket!)

## Jackal

Az örökélet beviteléhez ■ **BASIC/CODE/47616** file-térképpel rendelkező verziót választottuk. Ne töltsünk be semmit, hanem írjuk be ■ következőket:

```
10 CLEAR 65535: FOR i=65064 TO 65090: READ a: POKE i,a: NEXT i
20 DATA 62,255,221,33,0,64,17,0,186,55,205,86,5,62,129,50,55,152,50,136,152,49,244,251,195,0,91
30 RANDOMIZE USR 65064
```

Futtassuk ■ programot, és töltsük be ■ 47616-os blokkot.

## Super Stunt Man

Az örökélet és ■ végtelen idő bevitelét arra ■ verzióra közöljük, amelynek file-térképe ...../35000/1494.

Töltsük be ■ loader-t, majd írjuk be a következő programot:

```
10 CLEAR 24999: LOAD"" CODE: LOAD"" CODE: POKE 27277,0: POKE 25517,0: RANDOMIZE USR 25000
```

Futtassuk a programot, és indítsuk el ■ magnetofont.



Az elmúlt két alkalommal megismertük a **SCREEN MACHINE** programcsomag első két részét, most a harmadik kezelését tekintjük át.

A **STRING MACHINE** olyan segédeszköz, amely mindenféle szöveget, grafikus karaktert és UDG elemet - beleértve attribútumait is - gépi kódú adattömbbé képes szervezni. Ezek az adattömbök közvetlenül visszahívhatók a képernyőre egy alkalmas ... **USR** utasítás segítségével. Lényeges a segítségükkel megvalósítható jelentős memóriamegtakarítás is, nem beszélve a nyomtatási sebesség megnövekedéséről.

A **STRING MACHINE** a szalagon egy 'st' nevű, 2713 byte hosszú BASIC betöltőből és egy 'stc' nevű, 1700 byte-os gépi kódú részből áll. Betöltés után megjelenik az alapmenü, amely a következő lehetőségeket kínálja a számunkra:

#### 1. CATALOGUE

A létrehozott karakterláncok listáját adja meg, természetesen elhelyezkedésükkel és hosszukkal együtt;

**V** - szövegrészletek válogatott, vagy teljes (all) megtekintését (**VIEW**) teszi lehetővé. Észrevehetjük, hogy az egyenkénti megtekintésnek nincs külön opciója az alapmenübe történő visszatéréshez, hiszen lehet, hogy az alsó sorba írni akarunk. Természetesen bármely billentyű megnyomásával visszakérülünk a menühöz.

**P** - a katalógust megjeleníti a nyomtatón. Fontos, hogy tartsuk számon a katalógus adta részleteket, mert sok munkánk kárba veszhet.

A katalógus egyszerre max. 48 tételt képes csak felvenni, többel nem célszerű próbálkoznunk!

#### 2. CONSTRUCT STRING

A szövegszerkesztés rendkívül egyszerű, magának a programnak a 5...100. sorait használjuk fel. Végül is egy olyan 'a\$' stringet hozunk létre, amely a szöveg egészét tartalmazza, a szükséges képernyő-pozíciókkal és attribútumokkal együtt. Ha a programot **RUN**-nal próbáljuk indítani, akkor kiírja a szöveget és további két opciót kapunk:

##### COMPILE - szerkesztés

**TOP SCREEN** - a felső 0-21. sorokba írhatunk

**BOTTOM SCREEN** - az alsó 22-23. sorokba írhatunk

**START ADDRESS** - meg kell adni a hozzáférhető legalacsonyabb szabad memóriahelyet

Ezt követően a katalógus látható az új részletekkel kiegészítve.

##### AMEND - javítás

#### 3. SAVE TO TAPE

Ennek az opciónak a segítségével kimenthetjük

a szalagra az összes stringet, amelyet katalógusba vittünk. Hozzárendeli a megadott állománynevet. A hely és hosszadatokat a katalógusból veszi: kezdőcím és kezdőcímhez képest számított hossz (utolsó string kezdőcíme + hossz - legelső kezdőcím)

#### 4. CLEAR TEXT

Kitakarítja a teljes listát, így hozzáláthatunk a következő szövegrész megalkotásához.

### A STRING MACHINE használata

A megszerkesztésre kerülő szöveget vigyük be a programba, az 5-100. sorokba. Ne zavarjon senkit, hogy valamennyi további programsor elnyomásra kerül a jobb áttekinthetőség érdekében.

Most állítsuk be közvetlen BASIC paranccsal a háttér színét (pl. **PAPER 0**, **ENTER**). Célszerű elutó színű keretet is beállítani.

Vigyük be tanulásképpen egy egyszerű stringet:

```
5 LET a$ = "HELLO"
```

Ha futtatjuk a programot, a "HELLO" megjelenik a képernyő bal-felső sarkában. Most nyomjuk meg az 'A' billentyűt (**AMEND** = javítás), ekkor visszakérülünk a programirási állapotba, s folytathatjuk:

```
10 LET a$ = a$ + "MATROZ"
```

Újra futtassuk a programot **RUN**-nal, láthatjuk, hogy a két string összefűződött, de nem a legyszerencsében, ezért a megszokott BASIC javítási módot felhasználva módosítsuk az 5. sort:

```
5 LET a$ = CHR$ 22 + CHR$ 5 +  
CHR$ 10 + "HELLO"
```

Már most felhívjuk mindenki figyelmét, hogy vigyázzunk! Ha nem a számunkra szabaddá tett 5-100. sorokat hívjuk le **EDIT**-tel, elszáll az egész munkánk.

Futtassuk újra a programot, látható, hogy mindaz, amit itt tettünk, a **PRINT AT 5,10;"HELLOMATROZ"** utasításnak megfelelő. A vezérlő karakterek részletesen áttekinthetők a *Spectrum* kézikönyvében, mi most csak emlékeztetőül idézzük fel a legfontosabb **CHR\$** parancsokat:

CHR\$	Jelentés	Megjegyzés
13	NEW LINE (ENTER)	sorhatároló
16	INK	kiírószín (0...9)
17	PAPER	háttérszín (0...9)
18	FLASH	villogás (0-1)
19	BRIGHT	fényesség (0-1)
20	INVERSE	inverz mód (0-1)
21	OVER	takarás mód (0-1)
22	AT	pozíció kód (CHR\$ 0...21 + CHR\$ 0...31)
23	TAB	tabuláció (0...31)



Ha ■ képernyő alsó két sorát is fel akarjuk használni, akkor ezt ■ két sort nullás és egyes sorként kezeljük, majd ■ végén a szerkesztésnél ■ helyükre kerülnek. Ne felejtjük el viszont, hogy *minden üzenet megjelenése törli az itt található információkat.*

**Néhány extra stringet is felhasználhatunk.** Az alapsmenühöz ■ következőt kell beírunk:

LET a\$ = "menu": GO TO 100

Ha ki akarunk törölni minden szöveget ■ memóriából, gépeljük ezt:

LET a\$ = "clear": GO TO 100

Próbáljuk meg összehasonlítani a hagyományos és **STRING MACHINE** által létrehozott string műveleteket az 'index' programblokk sorain keresztül:

5 PRINT AT 0,3; INK 3; BRIGHT 1; "SCREEN MACHINE"

10 PRINT AT 6,6; INK 6; BRIGHT 1; "SCREEN MACHINE ONE"; AT 8,9; INK 5; "Screen\$ Compressor"

15 PRINT AT 11,6; INK 3; BRIGHT 1; "2 SCREEN MACHINE TWO"; AT 13,9; INK 5; "Screen\$ Processor"

20 PRINT AT 16,6; INK 3; BRIGHT 1; "3 STRING MACHINE"; AT 18,9; "Text Compiler"

25 PRINT#0; AT 1,4; INK 4; FLASH 1; BRIGHT 1; "Select Program to Load"

Mindaz kb. 510 byte memóriát igényel, ám ha ezt ■ **STRING MACHINE** segítségével hozzuk létre, úgy csupán 253 byte lesz a hossza és jóval gyorsabban megjelenik ■ képernyőn.

Ha olyan szöveget akarunk megjeleníteni, amely UDG karaktereket is tartalmaz, ezeket a karaktereket ■ **STRING MACHINE** behívása után helyezzük el a memóriában, pontosan ■ szöveg listázásakor. Természetesen, ha az UDG karakterek is jelen vannak, akkor kevesebb tárolóhelyünk lesz valamivel ■ memória felső területén, azaz csak alig észrevehető módon foglal el területet.

### A stringek kiíratása

A **RAMTOP**-ot soha ne felejtjük el lejjebb állítani a gépi kód helybiztosítása miatt: **CLEAR** kezdőcím - 1, majd hívjuk be a szövegállományt: **LOAD "fílenév" CODE, s** ■ szövegünk bármelyik string kezdőcímére kiadott **RANDOMIZE** **USR...** utasítással megjeleníthető.

### A stringek áthelyezése

Amennyiben szükséges, a stringek át is helyezhetők ■ memória más területére, azonban ehhez ■ manipulációhoz **néhány POKE** utasítás is szükséges.

Példánkban tételezzük fel, hogy egy 120 byte

hosszúságú stringet kell áthelyezni a 37000. címről 28500-ra.

Az eljárás a következő lesz:

CLEAR 28499: FOR i=0 TO 119: POKE (28500 + i),PEEK (37000 + i): NEXT i

Ezt követően ■ gépi kód mutatóinak átigazítására az új kezdőcím az új hely + 16 értékű lesz mindig, azaz:

POKE (új cím + 6), (új cím + 16) - INT (új cím + 16/256) \* 256

POKE (új cím + 7), INT (új cím + 16/256) : RANDOMIZE USR (új cím)

Ezt az eljárást sorban meg kell tennünk minden stringre, esetleg egy rövid BASIC programot is írhatunk, hogy ■ művelet automatizálva legyen.

### A SCREEN MACHINE Microdrive-on

A felhasználók mintegy 15 %-a rendelkezik Microdrive-val, jogos az igényük, hogy ne hagyjuk figyelmen kívül ezt a tényt. Nos ■ **SCREEN MACHINE** program is alkalmassá tehető ■ Microdrive-val együtt történő felhasználásra, ehhez kövessük az itt következő információkat:

#### 1. Az egyes programblokkok futását állítsuk le

index	- BREAK
S.M.ONE	- load + BREAK
S.M.TWO	- load + BREAK
STRING MACHINE	- editor + USR 34007

#### 2. Hajtsuk végre ■ módosításokat

index: módosítsuk ■ 800...999. sorokat

S.M.ONE: módosítsuk ■ 2010., 3020., 4050. és ■ 9999. sorokat

SAVE "m";1;"sm1" LINE 9999

SAVE "m";1;"sm1c" CODE 28000,2300

S.M.TWO: módosítsuk ■ 350., 610., és ■ 999. sorokat

SAVE "m";1;"sm2" LINE 999

SAVE "m";1;"sm2c" CODE 26217,5000

STRING MACHINE: módosítsuk a 7100. és a 9999. sorokat

SAVE "m";1;"st" LINE 9999

SAVE "m";1;"stc" CODE 33300,1700

DEMO: módosítsuk a 998. és a 999. sorokat

Reméljük, hogy most már minden lényeges elemét kivesszük ennek ■ nem is olyan új, de igen hatékony programcsomagnak. Annyit még mindenképpen meg kell említenünk, hogy ■ programcsomaghoz tartozik egy demonstrációs program (demo) is, amely szemlélteti ■ három rész-programmal megvalósítható lehetőségeket.

További jó munkát, felhasználást kívánunk!



## POKE beviteli technika

Mint tudjuk programjaink többsége nem az eredeti gyári file formátumban szerepel kazettáinkon, s ezek igen nagy százaléka azonos file-térképet mutat:

pl. 146/6916/20000/20536/1704, vagy  
164/6912/20000/20536/6916/1704.

Nos itt említett file térképű programok az ún. SPEC-MATE cartridge-dzsel lettek lementve. A SPEC-MATE az AT&Y angol cég terméke, működésében hasonlít a MULTIFACE-hez, de nem tömöríti a memóriát.

Most egy POKE beviteli módszert ismertetünk, amely lehetővé teszi, hogy az ilyen formátumban rendelkezésünkre álló programba könnyen bevihessük POKE-jainkat.

A SPEC-MATE által lementett program két leglényesebb része: 20000/20536. Az első blokk a 25000-44999 memóriaterületre töltődik, míg a második a 45000-65535 közötti területre. Itt most két módszert ismertetünk. Az egyik módszer arra az esetre vonatkozik, ha a beolvasandó POKE a 25000-44999 memóriaterületre mutat, a második módszer pedig értelemszerűen a másik memóriaterületet célozza meg.

### 1.módszer:

Írjuk be a következő kis programot:

```
10 CLEAR 24999
20 FOR I = 23296 TO 23308
30 READ A: POKE I,A: NEXT I:
```

RANDOMIZE USR 23296

```
40 DATA 221,33,168,97,17,32,78,62,255,
55,195,86,5
```

Futtassuk a programot, ekkor a program készen áll arra, hogy ráolvassuk a SPEC-MATE formátummal lementett program 20000 byte-os részét. Amikor a számítógép kilírja az OK üzenetet, gépeljük be a megfelelő POKE-t, majd vegyük fel kazettán az eredeti helyére a 20000 byte-os részt:

SAVE "1" CODE 25000,20000

Nagyon fontos, hogy a magnetofon felvétel gombját csak a fejléc után nyomjuk le, azaz a kódot fejléc nélkül mentsük ki.

A játékot visszatöltve már működni fog a POKE.

### 2.módszer:

A teendő hasonló, mint az első esetben, csak most a 20536 byte hosszú részt kell betöltenünk, majd a POKE beírása után SAVE "2" CODE 45000,20536.

```
10 CLEAR 24999
20 FOR I = 23296 TO 23308
30 READ A: POKE I,A: NEXT I:
RANDOMIZE USR 23296
40 DATA 221,33,200,175,17,56,80,62,
255,55,195,86,5
```

Természetesen biztonsági okokból célszerű a POKE által módosított blokkokat külön kazettára felvenni, ill. az új programot külön kazettán összerakni.

## Outcast

Az örökélet bevitelét a 146/6916/20000/20536/1704 file-térképes programhoz készítettük el.

Az örökélethez az 57226-os címre kell zérust beírunk.

Töltsük be MERGE-dzsel a loader-t, és írjuk be a következőket:

```
20 REM 00000 (ENTER), majd POKE 23757,83 (ENTER)
20 CLEAR 24999: FOR i=23823 TO 23832: READ a: POKE i,a: NEXT i: RANDOMIZE USR 23760
30 DATA 175,50,138,226,61,55,229,195,86,5
```

Futtassuk a programot, és indítsuk el a magnetofont.

## Slap Fight

Az örökélet beviteléhez a BASIC/6912/40036 file-térképpel rendelkező verziót választottuk ki.

Töltsük be a loader-t, RESET, majd írjuk be a következőket:

```
10 CLEAR 24999: LOAD "CODE 16384: POKE 23739,111: LOAD "CODE: POKE 48873,182:
RANDOMIZE USR 48400
```

Futtassuk az új loader-t, majd indítsuk tovább a magnetofont.



## 1. Szinskála demonstráció

```

10 DIM c$(99,6): DIM c(7): LET g$="M": LET s$=""
  LET i$=CHR$(16): LET p$=CHR$(17): LET
  b$=CHR$(19)
20 FOR i=1 TO 10: LET g$=g$+g$: LET s$=s$+
  s$: NEXT i
30 LET g$=g$( TO 704)' LET s$=s$( TO 704)
40 FOR i=0 TO 7 STEP 2: POKE USR "M"+i,85:
  POKE USR "M"+i+1,170: NEXT i
50 FOR i=1 TO 7: READ c(i): NEXT i
60 FOR i=1 TO 6: RESTORE 460
70 FOR j=0 TO 9: LET k=i*10+j
80 GO SUB 420
90 NEXT j: NEXT i
100 FOR j=0 TO 9: READ a,b
110 LET c$(70+j)=i$+CHR$(a)+p$+CHR$(b)+
  b$+CHR$(0)
120 LET c$(80+j)=i$+CHR$(a)+p$+CHR$(b)+
  b$+CHR$(1)
130 LET c$(90+j)=i$+CHR$(b)+p$+CHR$(a)+
  b$+CHR$(1)
140 NEXT j: NEXT i
150 FOR k=1 TO 6: RESTORE 590
160 GO SUB 420: NEXT k
170 FOR k=7 TO 9: GO SUB 420: NEXT k
180 BORDER 0: PAPER 0: CLS
190 INK 7: PRINT AT 0,10:"SZINSKALA:"
200 PRINT AT 2,8:"0 1 2 3 4 5 6"
210 FOR i=1 TO 6: PRINT AT i*2+1,2: PAPER c(i):
  "-----":AT i*2+2,2:"-----":AT i*2+2,5: PAPER 0;:
220 FOR k=1 TO 2: PRINT AT i*2+k,7;
230 FOR j=0 TO 2: PRINT c$(i*10+j):"MM": NEXT j
240 PRINT c$(i*10+3):"---": PAPER 0:"---"
250 FOR j=4 TO 6: PRINT c$(i*10+j):"MM":
260 NEXT j: NEXT k: NEXT i
270 PRINT AT 3,24:"7 8 9"
280 FOR i=1 TO 5: FOR k=2 TO 3: PRINT AT i*2+
  k,23;
290 FOR j=7 TO 9: PRINT c$(i*10+j):"MM": NEXT j
300 NEXT k: NEXT i
310 FOR i=7 TO 9: LET a$="MM"
320 IF i=9 THEN LET a$="---"
330 PRINT AT i*2+2,2:c$(i):a$:AT i*2+3,2:a$:AT
  i*2+3,5: PAPER 0: INK 7: BRIGHT 0;:
340 FOR k=2 TO 3: PRINT AT i*2+k,7;
350 FOR j=0 TO 3: PRINT c$(i*10+j):"MM": NEXT j
360 PRINT PAPER 0:"---"
370 FOR j=4 TO 6: PRINT c$(i*10+j):"MM": NEXT j
380 PRINT PAPER 0:"---"
390 FOR j=7 TO 9: PRINT c$(i*10+j):"MM": NEXT j
400 NEXT k: NEXT i
410 GO TO 410
420 READ a,b,c
430 LET c$(k)=i$+CHR$(a)+p$+CHR$(b)+b$+
  CHR$(c)
440 RETURN
450 DATA 2,6,4,5,1,3,0
460 DATA c(i),0,0
470 DATA 0,c(i),1

```

```

480 DATA c(i),c(i),0
490 DATA C(i),c(i),1
500 DATA c(i),7,0
510 DATA 7,c(i),1
520 DATA c(i),7,1
530 DATA c(i+1),c(i),1
540 DATA c(i+1),c(i),0
550 DATA c(i),c(i+1),1
560 DATA 5,3,2,5,2,1,2,3
570 DATA 6,5,6,1,4,1
580 DATA 2,4,6,3,4,3
590 DATA c(k),c(k),0
600 DATA 0,7,0
610 DATA 7,7,0
620 DATA 0,7,1
630 LET a$="-"+CHR$(17)+CHR$(2)+"-"+CHR$(
  17)+CHR$(4)+"-"+CHR$(17)+CHR$(1)+"-": SAVE
  a$ LINE 10

```

## 2. Memória státusz

```

9996 DEF FN p(n) = PEEK n + 256*PEEK
  (n+1)
9997 CLS: DRAW 255,0: DRAW 0,175:
  DRAW -255,0: DRAW 0,-175: PLOT 40,
  158: DRAW 160,0: PRINT AT 1,6:
  "MEMORIA STATUSZ:";AT 4,1:"RAM memoria:----
  ---";(FN p(23732)-16383)/1024;
  "-K";AT 6,1:"Microdrive terkep: ";FN
  p(23631)-23734:" byte";AT 8,1:"BASIC
  program:-----";FN p(23627)-FN
  p(23635):" byte";AT 9,1:"Változok:-----
  ---";FN p(23641)-FN p(23627)-1:" byte";AT
  11,1:"Szabad memoria:---";FN p(23730)-
  FN p(23641):" byte";AT 19,1:"UDG:---
  ABCDEFGHIJKLMNOPQRSTU"
9998 PRINT AT 20,8:"ABCDEFGHIJKLMN
  MNOPQRSTU"
9999 PRIINT AT 14,1:"Time:-----";INT
  ((FN p(23672) + 65536*PEEK 23674)/
  50):"-mp.": GO TO 9999

```

**Megjegyzés:** A SPACE-ek helyét " karakterrel jelöljük.

**Az első program** a képernyőn szemlélteti a számunkra a lehetséges színből kikeverhető színnyalatok teljes arzenálját.

**A második program** kijelzi a mindenkor aktuális memória-státuszt. Célszerű a BASIC program végén elhelyezni, program írása közben a RUN 9996 utasítás kiadásakor megjelenik a képernyőn sorban a RAM státusz, a Microdrive státusz, az aktuális BASIC program hossza, a változóterület hossza, a még rendelkezésre álló szabad RAM memória nagysága, az UDG karakterkészlet képe és a bekapcsolás óta eltelt idő.



# A hanggenerátor programozása

Sokan megijednek, amikor elővesszük a gépi kódot, pedig eddig még csak az alapokat ismertük meg a hanggenerátor programozásához, természetesen eddigi ismereteinket már előnyösen kamatoztathatjuk ■ továbbiakban.

Vágjunk tehát bele: ne ijedjünk meg, nem lesz ez olyan nehéz, meglátjuk megéri. Az előző mintapéldánk vezérfonalán küsszunk tovább. Csak a vezérlő szakaszt kell átírunk gépi kódra, a DATA sorok könnyen átolvashatók a memória bármely területére.

A gépi kódú vezérlő programot ■ 30000. címtől helyezzük el, hossza 100 byte. Az assembly lista - mely egyénekenként változó módon bevihető a memóriába - ■ következő:

```

30000 33,172,117 LD HL,30124
;HL-ben tároljuk az adatok kezdőcímét
30003 126 LD A,(HL)
;A soron következő adat vizsgálata
30004 245 PUSH AF
;Az adatot eltesszük a verembe
30005 238,17 XOR 17
;17-es időzítési konstans ?
30007 32,4 JR NZ,30013
;Ha nem, ugrás tovább
30009 6,240 LD B,240
;Külső időzítő ciklus beállítása
30011 24,94 JR 30107
;Ugrás az időzítő rutinra
30013 241 POP AF
;Visszaolvassuk az adatot,
30014 245 PUSH AF
;és ismét eltesszük a verembe.
30015 238,18 XOR 18
;18-as időzítési konstans ?
30017 32,4 JR NZ,30023
;Ha nem, ugrás tovább
30019 6,120 LD B,120
;Külső időzítő ciklus beállítása
30021 24,84 JR 30107
;Ugrás az időzítő rutinra
30023 241 POP AF
;Visszaolvassuk az adatot,
30024 245 PUSH AF
;és ismét eltesszük a verembe.
30025 238,20 XOR 20
;20-as időzítési konstans ?
30027 32,4 JR NZ,30033
;Ha nem, ugrás tovább
30029 6,60 LD B,60
;Külső időzítő ciklus beállítása
30031 24,74 JR 30107
30033 241 POP AF
;Visszaolvassuk az adatot,
```

```

30034 245 PUSH AF
;és ismét eltesszük a verembe.
30035 238,16 XOR 16
;Adatmező vége ?
30037 32,15 JR NZ,30054
;Ha nem, ugrás tovább
30039 62,7 LD A,7
;Kimeneti csatornák lezárása
30041 1,253,255 LD BC,65533
30044 237,121 OUT (C),A
30046 62,63 LD A,63
30048 1,253,191 LD BC,49149
30051 237,121 OUT (C),A
30053 201 RET
;Zene vége
30054 241 POP AF
;Az adat a kijelölt regiszter számát
30055 1,253,255 LD BC,65533
;tartalmazza, a regisztert meghívjuk.
30058 237,121 OUT (C),A
30060 35 INC HL
;Az adatmutatót növeljük
30061 126 LD A,(HL)
;Beolvassuk a következő adatot,
30062 1,253,191 LD BC,49149
30065 237,121 OUT (C),A
;és beírjuk a megfelelő regiszterbe.
30067 62,13 LD A,13
;Burkológörbe-alak kiválasztása
30069 1,253,255 LD BC,65533
30072 237,121 OUT (C),A
30074 62,9 LD A,9
30076 1,253,191 LD BC,49149
30079 237,121 OUT (C),A
30081 35 INC HL
;Az adatmutatót tovább léptetjük
30082 62,127 LD A,127
;SPACE megnyomására a kimeneti
30084 219,254 IN A,(254)
;csatornákat lezárjuk, és kilépünk
30086 230,31 AND 31
;a rutinból.
30088 254,30 CP 30
30090 32,167 JR NZ,30003
;Egyébként ugrás vissza az új adat
30092 62,7 LD A,7
;vizsgálatához.
30094 1,253,255 LD BC,65533
30097 237,121 OUT (C),A
30099 62,63 LD A,63
30101 1,253,191 LD BC,49149
30104 237,121 OUT (C),A
30106 201 RET
30107 241 POP AF
;Páros PUSH/POP biztosítása
```



```

30108 197      PUSH BC
;A külső ciklus értékét kimentjük a
30109 6,0      LD B,0
;verembe, és betöltjük a belső ciklus
30111 245      PUSH AF
;értékét. Ezután 4 önmagában jelen-
30112 241      POP AF
;tételen veremművelet következik,
30113 245      PUSH AF
;ezek ciklikus végrehajtása eredmé-
30114 241      POP AF
;nyezi az időtagot.
30115 16,250    DJNZ 30111
;Belső ciklus
30117 193      POP BC
;A külső ciklus értékének visszaírása
30118 16,244    DJNZ 30108
;Külső ciklus
30120 35        INC HL
;Az adatmutató tovább lép
30121 195,51,117 JP 30003
;Ugrás vissza a következő adat viz-
;gálatához.

```

Ez a gépi kódú rutin bármilyen, az előzőekben megismert adatsor lejátszására alkalmas. Az adatsor kezdőcíme a **30124. cím**. Próbáljuk ki, hogyan szólnak gépi kódban a dallamok. Az első (kánon) mintapéldában írjuk be a következő sort:

```

90 FOR i=30124 TO 30594: READ a:
POKE i,a: NEXT i

```

Futtassuk ezt a sort (**RUN 90**), és ha a gépi kódú ru-

tin (30000-30123) is a memóriában van, bátran adjuk ki: **RANDOMIZE USR 30000 (ENTER)**  
A rutinból a **SPACE** billentyű megnyomásával bármikor kiléphetünk.

A második példánkat is hasonlóképpen átalakíthatjuk:

```

90 FOR i=30124 TO 31727: READ a:
POKE i,a: NEXT i

```

Beolvasás után a gépi kódú dallamok elővarázsolása hasonlóképpen történik.

Ha összehasonlítjuk a BASIC és a gépi kódú verziót, megállapíthatjuk, hogy az utóbbiban minden hang a helyén van, nem jelentkezik a BASIC lassúságából eredő elcsúszások, késések. Sőt, a gépi kód felhasználása lehetőséget ad az ütemek finomítására is. Most 3 időtag kapott helyet a programban: a **30010. címen 240-et**, a **30020. címen 120-at**, míg a **30030. címen 60-at** helyeztünk el. Írjunk most ezekre a címekre sorban 40-et, 20-at és 10-et, majd futtassuk ismét a rutint: **RANDOMIZE USR 30000**, meglátjuk jelentősen felgyorsítottuk a zene lejátszását.

Az elmúlt alkalomkor ismertetett módszer felhasználásával saját dallamokat is komponálhatunk, pl. kotta felhasználásával, majd az így elkészült adatsort fűzzük be a **30124. címtől**, és futtassuk saját programunkat is. Egyet ne felejtünk el, az adatsornak mindig 16-os adatbyte-tal kell záródnia, ezzel azonosítja a rutint a zene végét.

## Zynaps

A sérthetlenség elérését szolgáló segédprogramot a BASIC/768/4096/40535 file-térképpel rendelkező verzióhoz készítettük el.

Töltsük be a BASIC loader-t, majd RESET, és írjuk be az új loader-t:

```

10 CLEAR 32767: LOAD ""CODE: POKE 64531,214
20 RANDOMIZE USR 64512
30 POKE 65115,15
40 FOR f=23311 TO 23323
50 READ a: POKE f,a: NEXT f
60 DATA 62,126,50,106,177,62,201,50,80,155,195,0,128
70 RANDOMIZE USR 65082
RUN, és a betöltést folytassuk az első CODE file-től.

```

## Challenge of the Gobots

Az örökélet bevételét a 146/6916/20000/20536/1704 file-térképes verzióra készítettük el.

Az örökélethez az 51329-es címre 0-t kell elhelyezni.

Töltsük be a MERGE-dzsel a program loader-ét, majd gépeljük be a következőket:

```

20 REM 00000 (ENTER), majd POKE 23757,83 (ENTER)
20 FOR i=23823 TO 23832: READ a: POKE i,a: NEXT i: RANDOMIZE USR 23760
30 DATA 175,50,129,200,61,55,229,195,86,5

```

Futtassuk a programot (RUN), és indítsuk el a magnót.



## 3. Szubrutin hívások (CALL utasítások)

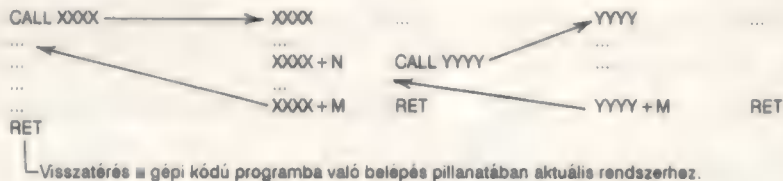
Ezeket a utasításokat viszonylag gyakran használjuk gépi kódú programjaink írásakor. Ha a BASIC nyelvvel vetjük össze a gépi kódot, akkor a esetben szerepe a GO SUB utasításnak megfelelő.

A CALL utasítás elvi mechanizmusát tekintsük meg egy egyszerű szemléltető ábra segítségével:

XXXX	CALL YYYY
XXXX + 3	következő utasítás
...	
YYYY	a szubrutin első utasítása
...	
YYYY + N	közbeneső utasítások
...	
YYYY + M	RET (a szubrutin lezárása)

Egy CALL utasítás kiadása esetén a programszámoló ~~számláló~~ értéke, vagyis a CALL utasításhoz tartozó alsó/felső byte-ot követő cím (XXXX + 3) a processzor verembe töltődik. Ezt követően a CALL utasítás mellett megadott cím (YYYY) betöltődik a programszámolóba, tehát a program futása itt meghatározott memóriacímen folytatódik. Amikor a szubrutinnak vége, a gép találkozik egy RET utasítással, a visszatérés a veremben eltárolt címhez fog történni (XXXX + 3), vagyis a töltődik a programszámolóba, és a futás itt folytatódik.

Szubrutinokat többszörösen is egymásba ágyazhatunk, ilyenkor a veremben folyamatosan eltárolódnak a visszatérési címek, az egyes RET utasítások a folyamatos visszatérést teszik lehetővé:



Az ide tartozó utasítások a következők:

CD X Y	205,x,y	CALL CIM	;feltétel nélkül
DC X Y	220,x,y	CALL C,CIM	;ha a átviteli jelzőbit 1
D4 X Y	212,x,y	CALL NC,CIM	;ha a átviteli jelzőbit 0
CC X Y	204,x,y	CALL Z,CIM	;ha a zérus jelzőbit 1
C4 X Y	196,x,y	CALL NZ,CIM	;ha a zérus jelzőbit 0
FC X Y	252,x,y	CALL M,CIM	;ha a előjel jelzőbit 1
F4 X Y	244,x,y	CALL P,CIM	;ha a előjel jelzőbit 0
EC X Y	236,x,y	CALL PE,CIM	;ha a túlcsoordulás/paritás jelzőbit 1
E4 X Y	228,x,y	CALL PO,CIM	;ha a túlcsoordulás/paritás jelzőbit 0

ahol x,y (Xh, Yh) a CIM alsó/felső byte-ja.

## 4. A RET utasítások

Az eddigiek ismeretében talán nem nehéz kitalálni, milyen célt szolgálnak ezek a utasítások. Szubrutinból való visszatérést tesznek lehetővé. Ha CALL utasítás után alkalmazzuk, úgy a CALL utasítást követő címen folytatódik a program futása. Ha nem használtunk CALL utasítást, vagy a CALL utasításokkal meghívott szubrutinokból már visszatértünk, a verem-mutató értékét a állítottuk el (pl. PUSH utasítással), a esetben a gépi kódú program meghívásának a színhelyére, a esetek többségében a BASIC operációs rendszerhez térünk vissza. A CALL utasításokhoz hasonlóan itt is megkülönböztetünk feltétel nélküli, ill. jelzőbit állapotától függő visszatérést.

Az ide tartozó utasítások a következők:

C9	201	RET	;feltétel nélkül
D8	216	RET C	;ha a átviteli jelzőbit 1
D0	208	RET NC	;ha a átviteli jelzőbit 0
C8	200	RET Z	;ha a zérus jelzőbit 1
C0	192	RET NZ	;ha a zérus jelzőbit 0
F8	248	RET M	;ha a előjel jelzőbit 1
F0	240	RET P	;ha a előjel jelzőbit 0
E8	232	RET PE	;ha a túlcsoordulás/paritás jelzőbit 1
E0	224	RET PO	;ha a túlcsoordulás/paritás jelzőbit 0

## 5. RST utasítások

Ezek az utasítások lehetőséget biztosítanak a ROM egyes területeire való közvetlen belépésre. Előnyük, hogy csak egy-byte-osak. Nem standard Z-80 utasítások, jelen formájukban csak a Spectrum esetében aktívak. Nyolc ilyen utasítást megkülönböztetünk meg, ezek a belépési pontok a ROM induló területén, a 0. címtől a byte-os emelkedéssel helyezkednek el.

Az ide tartozó utasítások a következők:

C7	199	RST 0000h (RST 0)	; RESTART rutin
Meghívása megfelel a RANDOMIZE USR 0 utasításnak.			
CF	207	RST 0008h (RST 8)	; hibacsatorna megnyitása

Segítségével saját rutinjainkban leágazásokat helyezhetünk el, hibauzenetek megjelenítéséhez.

Pl:

30000	CF	207	RST
30001	0A	10	DEFB 10

esetén a RANDOMIZE USR 30000 utasítás kiadásakor megjelenik az 'Integer out of range' hibauzenet.

D7	215	RST 0010h (RST 16)	; a PRINT csatorna megnyitása
----	-----	--------------------	-------------------------------

Segítségével karakterek megjelenítését oldhatjuk meg a képernyőn.

Pl:

30000	3E 02	62,2	LD A,2
30002	CD 01 16	205,1,22	CALL 5633
30005	3E 41	62,65	LD A,65
30007	D7	215	RST 16
30008	C9	201	RET

esetében, a RANDOMIZE USR 30000 utasítás kiadását követően meg fog jelenni a képernyő felső részén a '65'-ös kódú 'A' karakter. Az 5633-as ROM belépési cím a képernyő-csatorna megadásához használjuk (képernyő felső része = 2), egyébként az egyes ROM főbelépési pontokat a későbbiekben fogjuk ismertetni.

DF	223	RST 0018h (RST 24)	; a COLLECT rutin meghívása
----	-----	--------------------	-----------------------------

Segítségével meghívhatjuk a COLLECT karakterelhívó rutint. Ha előzetesen a CH ADD rendszerváltó által kijelöltünk egy memóriacímét, úgy a rutin megvizsgálja, hogy az a címen nyomtatható karakter van-e. Ha igen, akkor ennek a kódja az 'A' regiszterbe kerül.

E7	231	RST 0020h (RST 32)	; a COLLECT NEXT CHAR rutin meghívása
----	-----	--------------------	---------------------------------------

Segítségével meghívjuk a COLLECT NEXT CHAR rutint. Az előzőhöz hasonló módon működik, de itt a CH ADD rendszerváltó által meghatározott címét követő rekesz kódja kerül az 'A' regiszterbe, már amennyiben nyomtatható karakterről van szó.

EF	239	RST 0028h (RST 40)	; belépés a lebegőpontos kalkulátorba
----	-----	--------------------	---------------------------------------

Segítségével matematikai műveletek elvégzésére nyílik lehetőség.

Pl:

30000	3E 02	62,2	LD A,2
30002	CD 01 16	205,1,22	CALL 5633
30005	01 02 00	1,2,0	LD BC,2
30008	CD 2B 2D	205,43,45	CALL 11563
30011	01 04 00	1,4,0	LD BC,4
30014	CD 2B 2D	205,43,45	CALL 11563
30017	EF	239	RST 40
30018	0F	15	DEFB 15 = ADD
30019	38	56	DEFB 56 = EXIT
30020	CD E3 2D	205,227,45	CALL 11747
30023	C9	201	RET

Az itt látható demonstrációs program futtatásakor (RANDOMIZE USR 30000) megjelenik a képernyőn a két megadott szám (2 és 4) összege = 6. Mi is történt itt valójában? Azt már látjuk, hogy igen sok része ismeretlen a listának. Az első két utasítással megnyitottuk a képernyő felső részének csatornáját, másképpen nem fog megjelenni láthatóan az eredmény. A BC regiszterpárban megadott értékek a 11563-as ROM rutin meghívásával tölthetők át a lebegőpontos verem tetejére. Az RST 40 utasítással belépünk a lebegőpontos kalkulátorba, majd az ezt követő adatbyte-ok a verem felső elemeivel végeznek el matematikai műveleteket. A 15-os kód az összeadást jelenti, azaz a verem két felső elemét (2,4) összeadja, a eredményt (6) visszatesszi a verem felső elemének helyére. Az 56-os kód a kilépést jelenti a lebegőpontos kalkulátorból. Ezt követően a 11747-es címen induló ROM rutin segítségével jeleníthetjük meg a képernyőn a lebegőpontos verem legfelső elemét (jelen esetben tehát a 6-ot). A lebegőpontos kalkulátorral, a mint említettük a ROM terület egyes lényeges belépési pontjairól még részletesen lesz szó.

F7	247	RST 0030h (RST 48)	; Helybiztosítás a munkaterületen
----	-----	--------------------	-----------------------------------

Ez az ún. MAKE BC SPACES rutin, amely különböző kalkulációkhoz helyet biztosít a munkaterületen. A lefoglalandó helyek számát előzetesen a BC regiszterpárban kell megadni.

FF	255	RST 0038h (RST 56)	
----	-----	--------------------	--

Ez a rutin lehetővé teszi, hogy maszkolt megszakítás alatt léptessük az órát és használjuk a billentyűzetet. A maszkolt megszakításról is a későbbiekben lesz szó.



# PÁLYÁZATI REJTVÉNY

**Vízszintes:** 1. A QUEST Microsoft

ware játékprogramja, amelyet több

angol bolthálózatban a 128+2 gép-

hez mellékelnek 15. Edward becéze

16. Népszerű mesehős 17. Nátrium

19. Rejtő 21. Napszak 22. Műveltető

képző 23. Hónap rövid 25. Német folyó

27. TUL betűi keverve 29. Fagy szlo-

vákul 31. NSZK folyó 33. Fonal márka

35. Megbízható 37. Görög-keleti szent-

kép 38. Hivatalba járok, ügy... 39. Tá-

pé egynemű betűi 40. ... Angeles 42.

Uruguayi és kubai gépkocsik nemzet-

közi jelzése 43. Mecseki fennsík 44.

Fix-tár (számítás-technikai kifejezés)

46. Szájhős németül 49. Rossz közep-

pel 50. Elíz betűi keverve 51. Francia

város 52. Aromái 53. Magyar család-

név 54. Ilyen rádió is van 55. Néró

egynemű betűi 56. Holland autók nem-

zetközi jelzése 58. A vízszintes 44. be-

tűi keverve 59. Napszak népiesen 61.

Csinos 63. Hamis 65. Ló másképpen

66. Japán pénzegység 68. Római csá-

sár volt 70. Férfinév 72. Ló betegség

74. Sörfalat 76. Líbiai légitársaság 77.

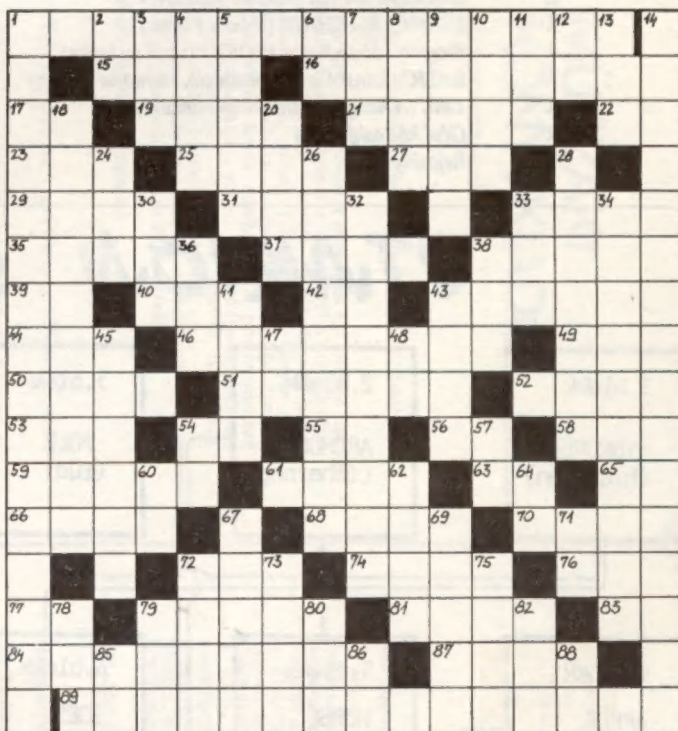
Repülőgéptípus 79. Csehszlovák áru-

házlánc 81. Apró mozgás 83. Gazdasági társulás

84. Kelta (bőség)isten 87. Rába betűi keverve

89. A

CABLE Software cég először a Mikro '87 kiállításon, Londonban mutatta be ezt a játékot



**Függőleges:** 1. A Gremlin Graphics ezzel a játékkal is bizonyította előkelő helyét a software piacon 2. Német munkaszolgálat a II.vh. idején 3. Világító dióda 4. Zenei kifejezés 5. Nyugi keverve! 6. Felkiáltó szó 7. Az első bárka tulajdonosa 8. Fiatal 9. Dán városka 10. Fogyaszthat 11. R.É.L. 12. Stroncium 13. Földművelő eszköz 14. A 'kód mesterei' által megjelentetett színvonalas szimuláció 18. Görög napilap 20. Orsolya becéze 24. Étkészlet része 26. ... Veikko, finn sífutó 28. Szakadást helyre hozó 30. Gépkocsi márka 32. USA város 33. Francia, spanyol, thaiföldi gépkocsik nemzetközi jelzése 34. ... Hermann, az 1800-as évek német filozófusa 36. Főtemplom 38. Takar 41. Svéd gk. márka 43. Italosan 45. Szakember 47. Puma közepe! 48. -on párja 54. Ez franciául 57. Szolmizációs hang 60. Vietnami gépkocsik nemzetközi jelzése 62. Ritkán használt szék 64. Alá 67. Heinrich ... (1797-1856) német író 69. Zenei műfaj 71. Római 1050 72. Vég nélküli Krudyl! 73. Nógrádi település 75. U.a. mint a vízsz. 31. 78. Népi Ellenőrzési Bizottság 79. Foszfór, nátrium 80. Nyers németül 82. Csup 85. A Spectrum Világ egyik szerkesztőjének névjele 86. Tagadószócska 88. A kettősség jelölése

Beküldendő a vízszintes 1., 89. valamint a függőleges 1., és 14. sorok. A helyes megfejtést beküldők közül 5 nyertesnek elküldünk egy-egy darab - általuk előre kiválasztott, és a megfejtéssel együtt megjelölt - SPECTRUM vagy COMMODORE programkassztát!

Megfejtéseket csak a SpV következő részének megjelenéséig fogadjunk el.

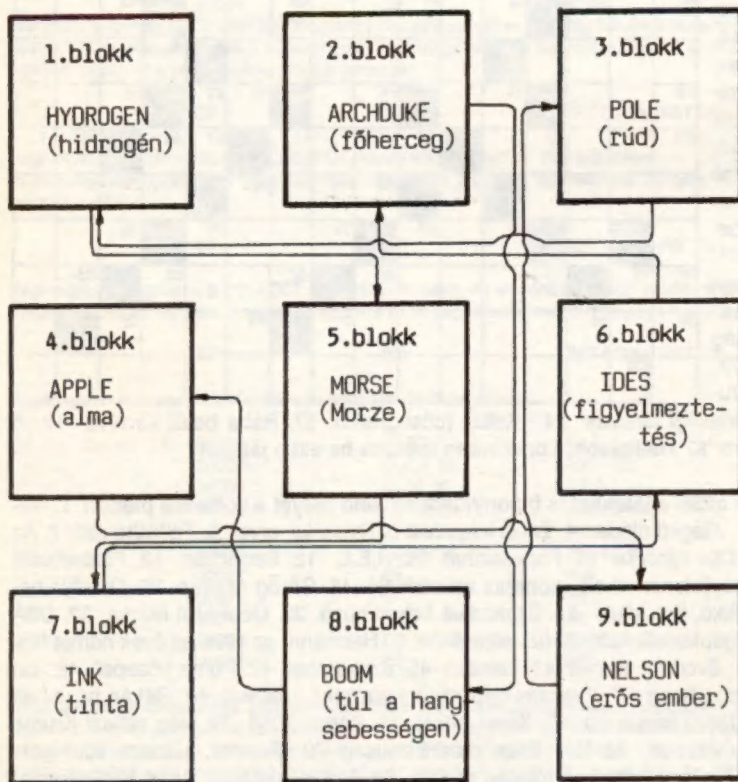
A nyereményeket postázzuk!



# Tartalomjegyzék

1	'SAM'-ünk előtt az új PC	1
2.	Játék, POKE, térkép	2
2.1	Northstar (Gremlin Graphics)	3
2.2	Garfield (The Edge)	5
2.3	Target Renegade (Imagine)	9
2.4	Dun Darach (Gargoyle Games)	10
3.	ENTERFACE (Enterprise melléklet)	15
4.	Ismeretlen nyelvek (Micro-PROLOG)	19
5.	STRING MACHINE (Print'n Plotter)	23
6.	Programozástechnika (POKE beviteli technika)	25
7.	BASIC (Színskála demonstráció, memória státusz)	26
8.	I28K (A hanggenerátor programozása)	27
9.	Gépi kód tanfolyam	29
+	Rejtvény	31

## STARION 9.zóna



Szállítmány

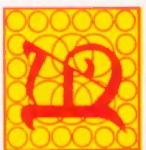
ARCHDUKE  
IDES  
NELSON  
BOOM  
HYDROGEN  
INK  
MORSE  
POLE  
APPLE

Blokk

1. A lelőtt Ferdinánd nemesi címe  
2. Cézár, vigyázz március 15-én  
3. Egy szívos ember meghal Trafalgar-nál  
4. A Concorde hangsebesség felett repül  
5. A Zepelin üzemanyaga  
6. Guttenberg túl fakón nyomtat  
7. Egy táviró berendezést tervezett  
8. Amudsen déli zászlótartója  
9. Newton meglátja a gravitáció gyümölcsét  
A 9. zóna kulcsszava: AMPHIBIAN (kétlábú lény)

Bizonyára sokan vannak, akik nem tudják, hogy a STARION játék ezzel a 9 zónával nem ér véget. A 9 zóna fő kulcsszavainak kezdőbetűiből is ki kell raknunk egy kódot. A kód a következő: AEROSPACE. A helyes kód megadását követően belépünk a 2.szintre, ahol újabb 9 zónával kell megbirkóznunk...





## MŰSZAKI KÖNYVKIADÓ

programajánlata:

**Patáki Attila - Tallér Ferenc: Fűtési rendszerek számítása személyi számítógéppel**

Vajon Ön gondolt-e már arra, hogy *othon is elvégzheti* egy készülő ház, társasház fűtési rendszerének számítógépes mértelezését? A szerzők által elkészített programcsomag alkalmas a *kémény-kerület, kémény-fúzat, vezeték ellenállás, nyomóvesztesség, blende-kétfélező, hőszükséglet és a fűtésiadomén* adatainak meghatározására. Az egyes számítási algoritmusok külön-külön program-blokkban lettek elhelyezve. A programcsomag használatát könyv is támogatja, ebbe mindenképpen hasznos beletáplozni, hiszen ipari jellegű felhasználók részére is hasznos információkat nyújt (pl. kazánház-kialakítás problémái, stb.). A számítógépes program jelenpillanatban **ZX-81, SPECTRUM 48K és COMMOORE 64** személyi számítógépekre áll rendelkezésre.

**Programcsomag: 340,- Ft** (minden géptípusra)  
**Könyv: 158,- Ft**

**Kepes János: Mikroszámítógépes grafika, grafikai algoritmusok**

A **SPECTRUM** grafikai „nyelvazete” az átlaghoz képest jó színvonalúnak mondható, ezt a lehetőséget ragadja meg a szerző is, hogy bevezesse a grafikai programozás iránt érdeklődőket ebbe az igéretes világba. A kazetta 'A' oldalán egyszerűbb algoritmusok telepítésével ismerkedhetünk meg. Ötleteteket kapunk a *labirintusok* építéséhez, *véletlen vonalak*, *pótlók-örvénycs*, valamint *átlagosító götöbök* rajzolásához, végül egyszerűbb ábra elkészítési tanulmányt meg. A kazetta 'B' oldalán már bonyolultabb algoritmusokkal találkozhatunk. Egy *rajzoló programot* készített a szerző, amelyen belül elérhető a *sötét és a rozsdás rajzolás* is. A rajzolóprogramot követően a grafikai programozás magasabb szintű kihasználására, az *animációra* kaphatunk ötleteket, szemléletes demonstrációs ábrák bemutatása mellett. Ezt a programcsomagot is támogatja egy könyv, amely további magyarázatokkal segíti a témában való elmélyedést.

**Programcsomag: 290,- Ft**  
**Könyv: 55,- Ft**



## MŰSZAKI KÖNYVKIADÓ

könyvajánlata:

**Kernighan - Pike: A UNIX operációs rendszer**

A **UNIX** a 16 bites számítógépek operációs rendszere, amely a nyolcvanas évekre kitépelt a nagyszámítógépek viszonylag kevéske számára hozzáférhető világából, s mind gyakorlatban jelenik meg a személyi számítógépeken is. Népszerűségét két tényezőnek köszönheti. A filozófiájának, amely "teljeskörűséget" két tényezőnek köszönheti, biztosit, és a hordozhatóságának, miután magasszintű programnyelven, *C-ben írták meg*.

A könyv nem felhasználói kézikönyv. Az a célja, hogy a **UNIX** használatában járatosan olvasói megismeresse a rendszer filozófiájával, elemelve és a **UNIX** alatti programozás technikájával. A cél elérését bőséges, a valós életből vett példamagy segíti, amelyben a különböző **UNIX**-változatok közös része szerepel.

**208 oldal, kb. 150,- Ft**

**Hanák Péter: ELAN0 kézikönyv és példatár**

Az **ELAN0** egy oktatási célú számítógépes programnyelv, amelyet a nyugati országokban már egy évtizede használnak, és 1981 óta a *Budapesti Műszaki Egyetemen* is sikeresen alkalmaznak. A könyv első része magát a nyelvet, a programozási környezetet ismereti, második részében pedig egyszerű, majd bonyolultabb példák között a játékok és a matematika köréből.

**312 oldal, 150,- Ft**

*A programok és könyvek megrendelhetők ill. megvásárolhatók:*

**MŰSZAKI KÖNYVKIADÓ**  
**Kándó Kálmán Könyvesboltja**  
Budapest, V. Bajcsy Zs. út 20. - 1051



# SPECTRUM EMULÁTOR



## A SINCLAIR ZX SPECTRUM

a személyi számítógépek között  
a 80-as évek **nagy slágere.**

Talán nem mindenki tudja, hogy  
szoftver kínálatát még a  
**COMMODORE 64**-esét is  
felmúlja!

Elsősorban ez az oka, hogy  
még ma is **hatalmas tábor** van  
ennek a gépnek.

A ZX rajongók érthetően  
nem szívesen mondanak le  
jól bevált programjaik  
használatáról...

Megszokta a **ZX SPECTRUM**-ot?

Továbbra is használni akarja  
megszokott gépét?

Ugyanakkor szeretne egy  
nagyobb teljesítményű számítógépet?

A megoldás:

**ENTERPRISE 128 +  
SPECTRUM emulátor!**

Kapható:

**CENTRUM Úttörő és Ifjúsági Áruház**

*Bp.V.Kossuth L.u.9.*

**CENTRUM Otthon Áruház**

*Bp.VII.Rákóczi u.74/76.*

**CENTRUM Kispesti Áruház**

*Bp.XIX.Kossuth tér 4/5.*

**Ára.: 7770,- Ft**

*(System bus bridge nélkül.: 6300,- Ft)*

